『グランツーリスモ７』における
リアルタイムレイトレーシング技術と
メソスケール材質表現

㈱ポリフォニー・デジタル

平井 健太朗

高野 修一

POLYPHONY™
DIGITAL

グランツーリスモ7

- PlayStation®4/5 用ドライビングシミュレーター
- より進化した挙動
- 400車種以上を再現
- フォトリアリスティックな映像
- 時間天候変化をサポート

POLYPHONY™ DIGITAL

本セッションの内容

- リアルタイムレイトレーシング
- 路面と芝生のメソスケール材質表現

POLYPHONY™ DIGITAL

リアルタイムレイトレーシング

高野 修一

POLYPHONY DIGITAL

モチベーション

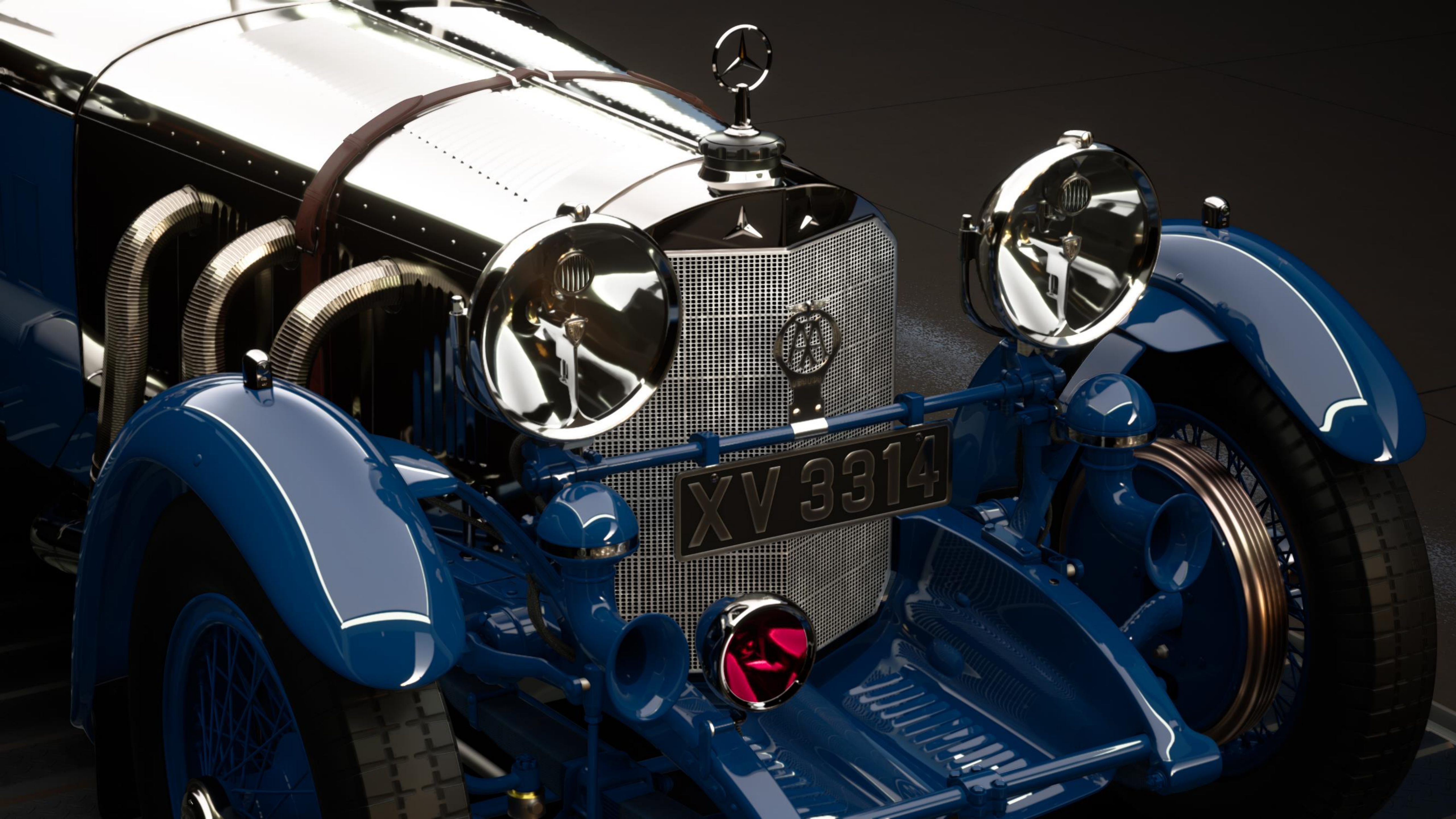
- PlayStation®5 でHWレイトレーシング機能をサポート
- クルマとレイトレーシングの相性の良さ
  - 艶のあるマテリアルで効果がわかりやすい
- より写実的な表現をしたい

POLYPHONY™ DIGITAL

レイトレーシング使用シーンの一例

POLYPHONY™ DIGITAL

AMG
GRAN TURISMO
Transurban


XV 3314

BMW
M

GT7の基本レンダリング構成

- Forward Renderer
  - 多彩なマテリアル
  - 帯域幅で有利

- 物理ベース
  - ？

物理ベースレンダリング

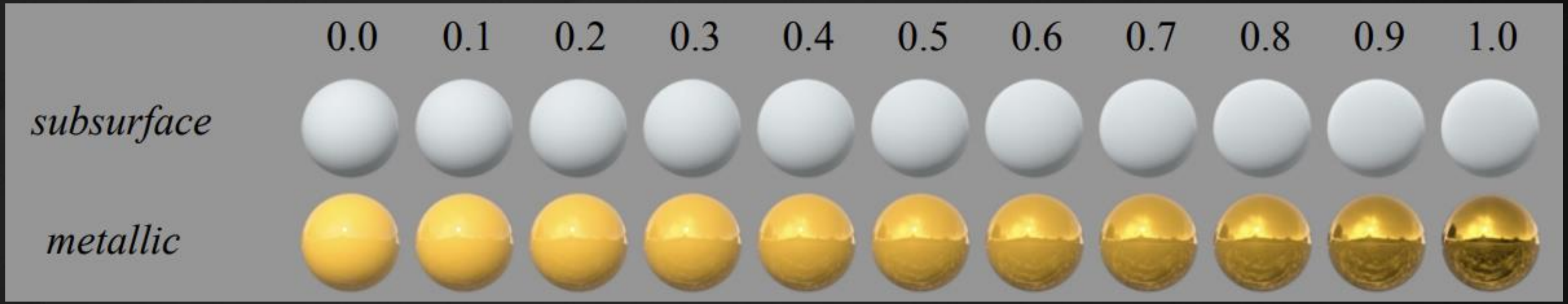
- 光の物理的性質
  - エネルギー保存
  - ヘルムホルツの相反性
  - フレネル反射
  - …

- 物理的性質を（概ね）満たす計算モデルを使う
- GTではDisney Principled BRDFを使用

POLYPHONY™ DIGITAL

Disney Principled BRDF

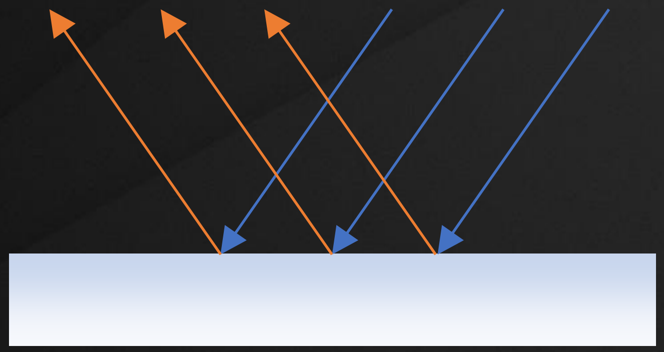
- SIGGRAPH2012
- アーティストに直感的な0-1の範囲のパラメータで構成
  - base color
  - roughness
  - …



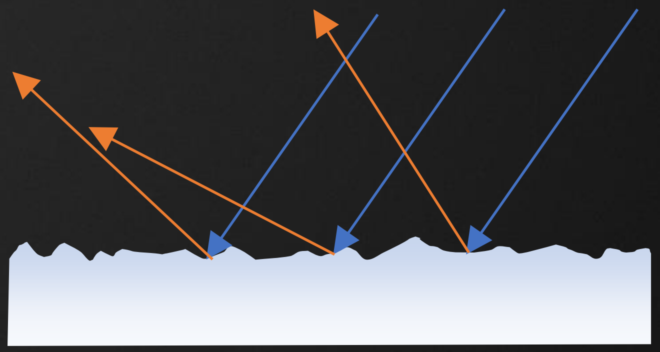
- パラメータセットを絞って簡略化したものを使用
- BRDFをスペキュラーとディフューズに分けて表現

スペキュラー

- 物体表面での鏡面反射成分
- 微小な平面の集合と考える(microfacet)
    - 法線の分布を microfacet分布関数で近似(DisneyはGGXを使用)
    - 荒さをroughnessで表す
- 微小面単位では理想的な鏡面反射が起こっている
- フレネル反射



完全鏡面

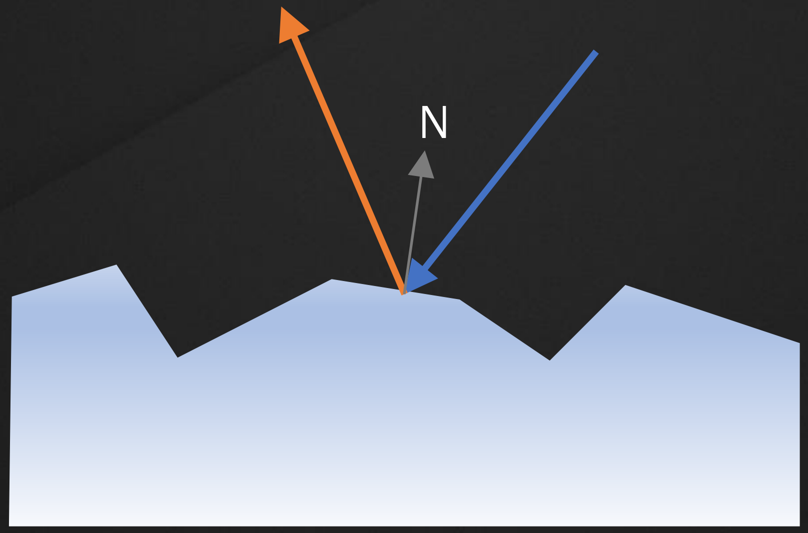


Glossy

POLYPHONY™ DIGITAL

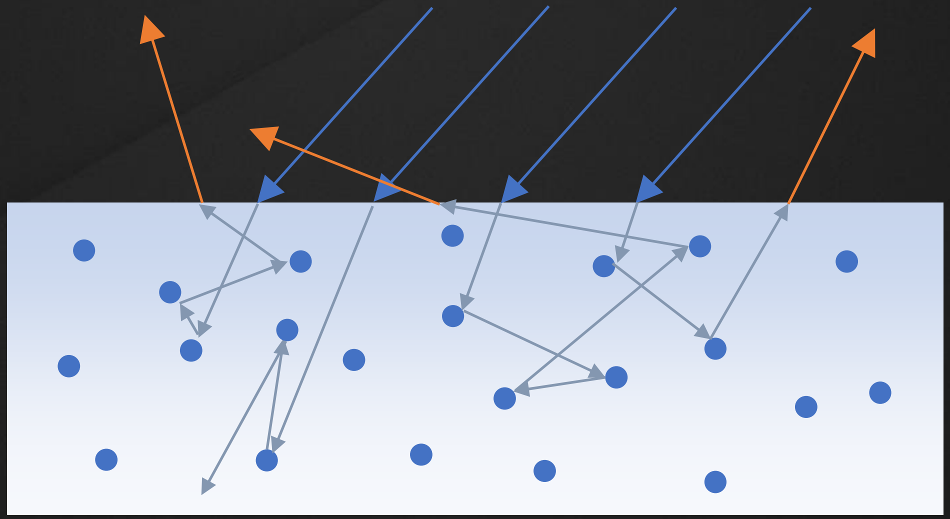
スペキュラー

- 物体表面での鏡面反射成分
- 微小な平面の集合と考える(microfacet)
  - 法線の分布を microfacet分布関数で近似(DisneyはGGXを使用)
  - 荒さをroughnessで表す
- 微小面単位では理想的な鏡面反射が起こっている
- フレネル反射



ディフューズ

- 物体表面から屈折して内部に侵入する光の成分
- 複雑な経路を経て表に再放射
- 全体として全方向に均等に拡散する



POLYPHONY™ DIGITAL

ライティング

- BRDFを元にレンダリング方程式を解く
  - $L(\omega_o) = \int f\,(\omega_i, \omega_o)\,L(\omega_i)(N \cdot \omega_i)d\omega_i$
  - $f(\omega_i, \omega_o) = f_{diffuse}(\omega_i, \omega_o) + f_{specular}(\omega_i, \omega_o)$
  - $L(\omega_i) = L_{local}(\omega_i) + L_{global}(\omega_i)$

- 解析光源の局所照明成分は解析的に解ける(ものもある)
  - 太陽を平行光源で近似など
- 大域照明(GI)成分は数値積分が必要
  - 太陽以外の空
  - 壁・オブジェクトからの反射光

POLYPHONY™ DIGITAL

大域照明のリアルタイム解法

- 事前積分キューブマップによる近似
- Screen Space Reflection(SSR)による近似
- レイトレーシング

POLYPHONY™ DIGITAL

従来のGTのディフューズライティング

- 従来の方法でも比較的近似精度が高い
  - 球面調和関数の事前ベイク(CEDEC2018)
  - Screen Space AO
  - ライトプローブ

POLYPHONY™ DIGITAL

ディフューズ

リファレンス(パストレ)

ラスタライズ

リアルタイムレイトレーシングは不要

従来のGTのスペキュラーライティング

- 従来のGTのレンダリングでは比較的近似精度が低かった
  - 事前積分キューブマップ(IBL)のみ
  - SSRなし
  - ローカルなオクルージョンの精度の低さが特に問題

- スペキュラー計算をレイトレーシングに置き換え

POLYPHONY™ DIGITAL

LaFerrari
レイトレーシングなし

レイトレーシング使用

レイトレーシングなし

レイトレーシング使用

POLYPHONY™ DIGITAL

McLaren
レイトレーシングなし

レイトレーシング使用

レイトレーシングなし

レイトレーシング使用

POLYPHONY™ DIGITAL

処理の流れ

- ラスタライズ
  - 解析光＋ディフューズGI
  - 薄いG-Bufferを出力

- レイトレーシングによるスペキュラーGI計算
  - G-Bufferからレイを発生
  - レイの衝突点からの放射輝度を計算
  - デノイズ

- 合成
- 半透明やポストエフェクトなど

POLYPHONY™ DIGITAL

ラスタライズパス

- 概ね通常と同じForwardレンダリングを行う
  - 解析光源のディフューズ、スペキュラー
  - ディフューズGI
  - スペキュラーGIは乗せない
  - カラー、ベロシティを出力

- いくつかのアトリビュートをG-Bufferに追加で出力
  - 法線
  - Roughness
  - スペキュラー係数

AMG
AMG

レイトレーシングパス

- スペキュラーGIを計算したい
- レイトレーシングAPIを使ってレンダリング方程式を解く

POLYPHONY™ DIGITAL

スペキュラー計算

- $L_{specular}(\omega_o)$
  $= \int D(\omega_i, \omega_o, \alpha) \cdot F(\omega_i, \omega_o, f_0) \cdot G(\omega_i, \omega_o, \alpha) \cdot L(\omega_i)(N \cdot \omega_i) d\omega_i$
  $\cong \left(\int F(\omega_i, \omega_o, f_0) \cdot G(\omega_i, \omega_o, \alpha)(N \cdot \omega_i) d\omega_i \right) \cdot \left(\int D(\omega_i, \omega_o, \alpha) \cdot L(\omega_i)\, d\omega_i \right)$
- ここで
  - D: 法線分布関数
  - F: フレネル項
  - G: マスキング関数
  - L: 放射輝度
- 簡単のためレイトレーシングでもこの近似を用いる

POLYPHONY™ DIGITAL

スペキュラー計算

- $\int D(\omega_i, \omega_o, \alpha) \cdot L(\omega_i)\, d\omega_i$ を計算したい

- モンテカルロ積分による離散化
- $\frac{1}{N}\sum\frac{D(\omega_i,\omega_o,\alpha)L(\omega_i)}{p(\omega_i)}$

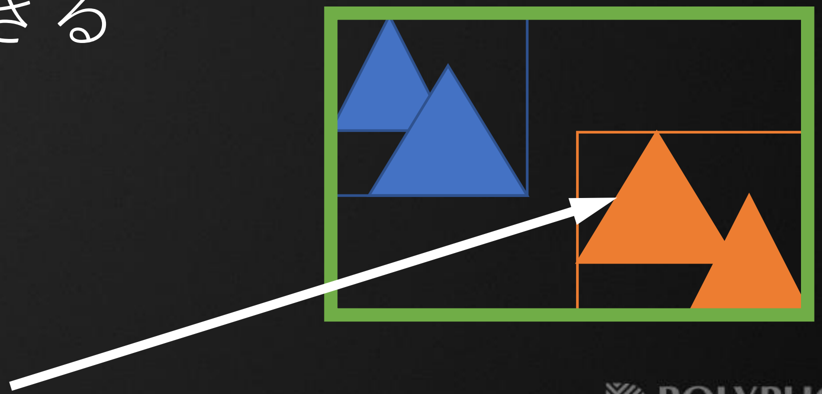
- レイトレーシングAPIを用いて計算

POLYPHONY™ DIGITAL

レイトレーシングAPI

- PSR
- 2レベルのアクセラレーションストラクチャ(BVH)
- 位置と方向を指定してレイキャスト
- 衝突のクエリができる

- 衝突した点の放射輝度を求める処理を行うことで、モンテカルロ積分の１サンプルを求めることができる



POLYPHONY DIGITAL

フレームあたりのサンプルの削減

- パフォーマンスの関係で多くのサンプルはできない
  - 少数のサンプルとデノイズで近似
  - 縮小バッファを使う

- 4K ⇨ 2Kの縮小バッファ
- 1 サンプル/縮小ピクセル

レイの生成

- ラスタライズでのG-Bufferを用いてレイを生成する
  - 起点: デプス、スクリーン座標
  - 方向: 法線、Roughness

- 方向計算にはDの確率密度を用いたインポータンスサンプリングを用いる
  - $\phi_h = 2\pi\xi_1$
  - $cos\theta_h = \sqrt{\frac{1-\xi_2}{1+(\alpha^2-1)\xi_2}}$
  - $h = (sin\theta_h cos\phi_h, sin\theta_h sin\phi_h, cos\theta_h)$
  - $p(h) = \frac{\alpha^2}{\pi\left((\alpha^2-1)\cos^2(\theta)+1\right)^2}$
  - ξ_1, ξ_2: ノイズ

- ワールド座標に変換してレイトレーシング！

POLYPHONY™ DIGITAL

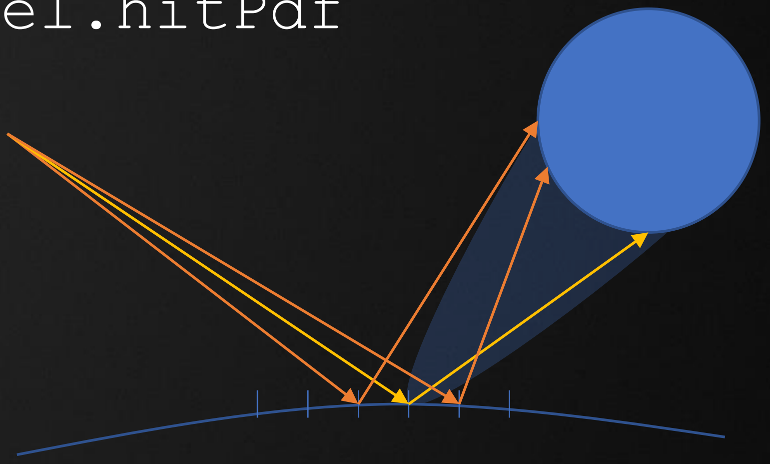
シェーディング

- レイが衝突した点から起点に向けた放射輝度を計算
- ラスタライズと同等のアルゴリズム
- 衝突した面のマテリアルの Function Shader 呼び出し

サンプル再利用フィルタ

- 周辺のサンプルを再利用して実効的なサンプルを増やす
- Stochastic Screen-Space Reflections [Stachowiak 2015]

```
result    = 0.0
weightSum = 0.0
for pixel in neighborhood:
    weight = localBrdf(pixel.hit) / pixel.hitPdf
    result += color(pixel.hit) * weight
    weightSum += weight
result /= weightSum
```



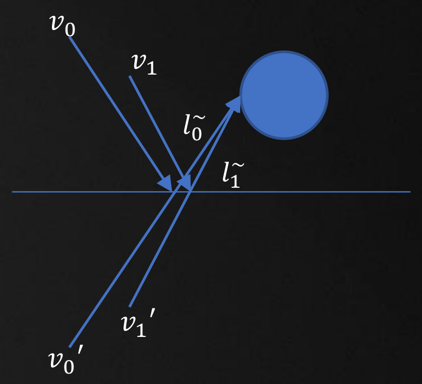
POLYPHONY™ DIGITAL

AMG

テンポラルフィルタ

- 前のフレームまでの計算結果を再利用
- 指数移動平均によるモンテカルロ積分

- ２種類のリプロジェクション
  - ベロシティによるリプロジェクション(roughness高)
  - 平均反射距離によるリプロジェクション(roughness低、平面)



- 過去のピクセルの輝度の分散などからウェイトを決定
  - exp(-scale * abs(history.x – colorAve.x) / sqrt(historyVariance))
- 現在のサンプルの色の分散から一定の範囲の色の外にあれば棄却
- Roughnessなどに応じてパラメータは調整...

- ヒストリバッファにYCoCgとY^2を出力して分散を計算

AMG

空間フィルタ

- 中央からの差分で重み付けされたバイラテラルフィルタ
  - 輝度差
  - 法線の差
  - デプス差
- 分散が高いピクセルの輝度差のウェイトを削減
  - ノイズ収束に伴いdetailが保持される
- roughness 等を使って係数を調整

POLYPHONY™ DIGITAL

AMG

テンポラルアンチエイリアシング

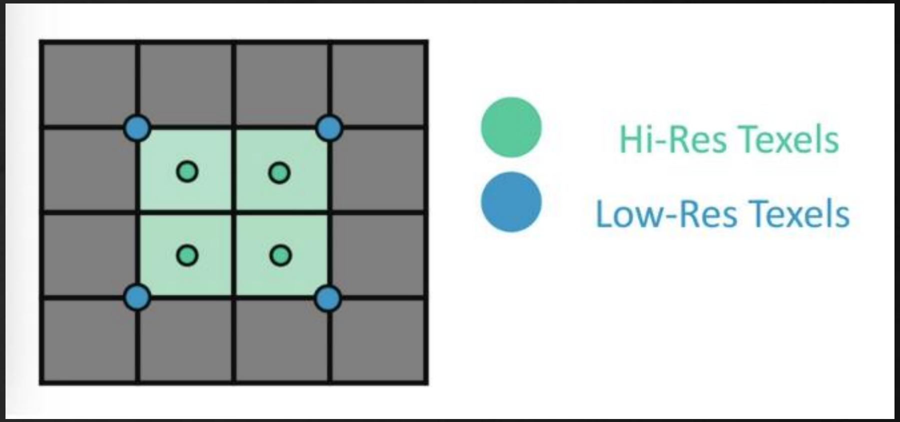
- レイトレーシング縮小バッファに対するTAA
- ベロシティによるリプロジェクション

- 前段のテンポラルフィルタとは別に必要
  - ピクセル分散を壊さない
  - 空間フィルタ後に適用することで安定に

- 劇的に安定化

合成

- バイラテラルアップサンプリング
  - 高解像度デプスと縮小デプスを見て重み付け



- スペキュラ係数バッファをかけてラスタライズパス出力と合成
- 空、半透明、ポストエフェクト等へ続く

POLYPHONY™ DIGITAL

AMG

AMG
AMG

制限

- 半透明部分へのレイトレーシングスペキュラーには非対応
- G-Bufferを別途描画することで対応は可能

まとめ

- 物理ベースレンダリングについての簡単な解説
- レイトレーシングを使ってスペキュラーGIを計算
- 複数のフィルタによるデノイズでサンプルを削減

POLYPHONY™ DIGITAL

今後

- 高速化
  - サンプリングパターン改良
- ノイズの削減
  - 光源分布を使ったインポータンスサンプリング
  - より良いフィルタアルゴリズム
- ディフューズ対応
- パストレーシング(リファレンス)

POLYPHONY™ DIGITAL

参考

- [Shopf 2009] Jeremy Shopf, Mixed Resolution Rendering
- [Burley 2012] Brent Burley, Physically-Bases Shading at Disney
- [Stachowiak 2015] Tomasz Stachowiak, Stochastic Screen-Space Reflections
- [Schied 2017] Christoph Schied, Real-Time Reconstruction for Path-Traced Global Illumination
- [Stachowiak 2018] Tomasz Stachowiak, Raytracing in Hybrid Real-Time Rendering
- PlayStation 5 SDK

路面・芝生の
メソスケール材質表現

平井　健太朗

POLYPHONY™ DIGITAL

路面・芝生の材質表現

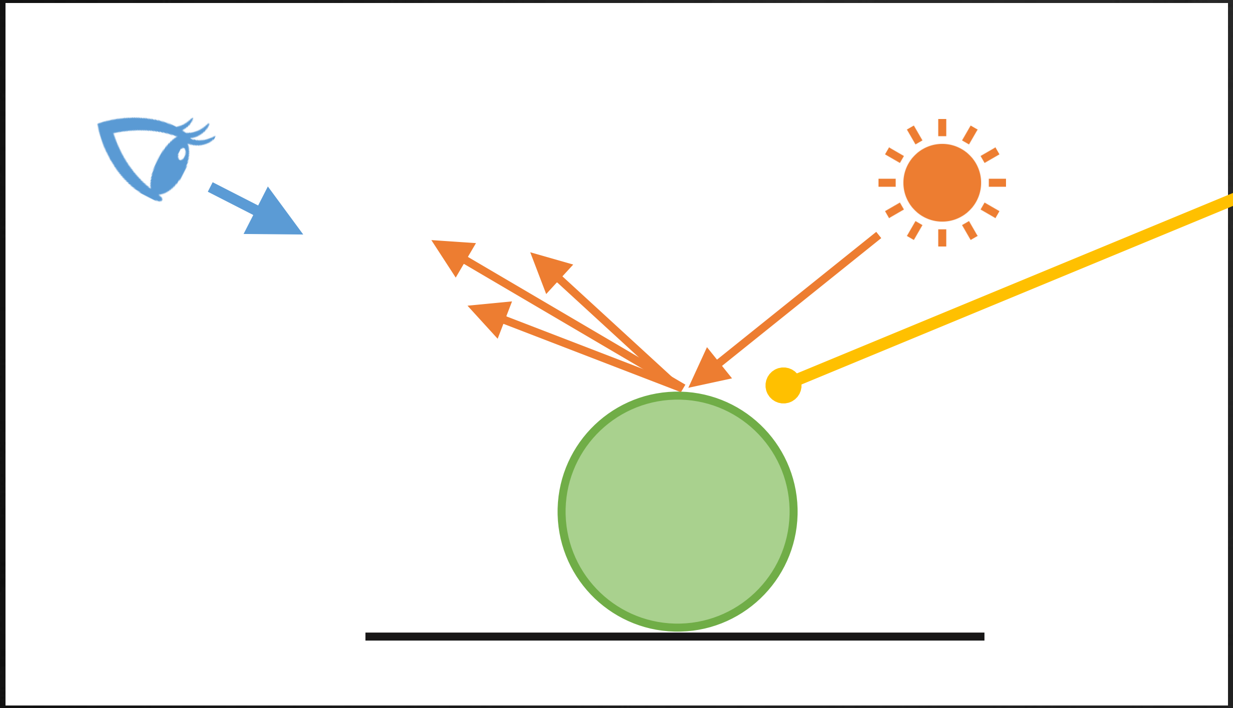
GT7では**路面・芝生**に対して専用のシェーダーを導入することになった。

- 画面面積の多くを占め、重要度が高い。
- 表面の凹凸による特徴的な見た目。

CGにおける材質表現

光の反射の仕方が見え方を決定する。

- 光の反射の仕方を関数化 (**BRDF**)
- 形状やパラメーターで材質を表現



入力

- 光源方向
- 視線方向
- 3Dモデル形状
- テクスチャ

…

出力

反射光の分布

- 方向
- バラつき
- 強さ（色）

POLYPHONY DIGITAL

シェーディングモデル

現実の現象をモデル化し、狙った見た目を再現する。

- 現実の光の反射を模倣した計算モデルを考える
  - 一般的にあまり重要でない部分は簡略化
    （軽量、汎用的なシェーディングモデルの作成）
  - 例外：スタイライズド表現

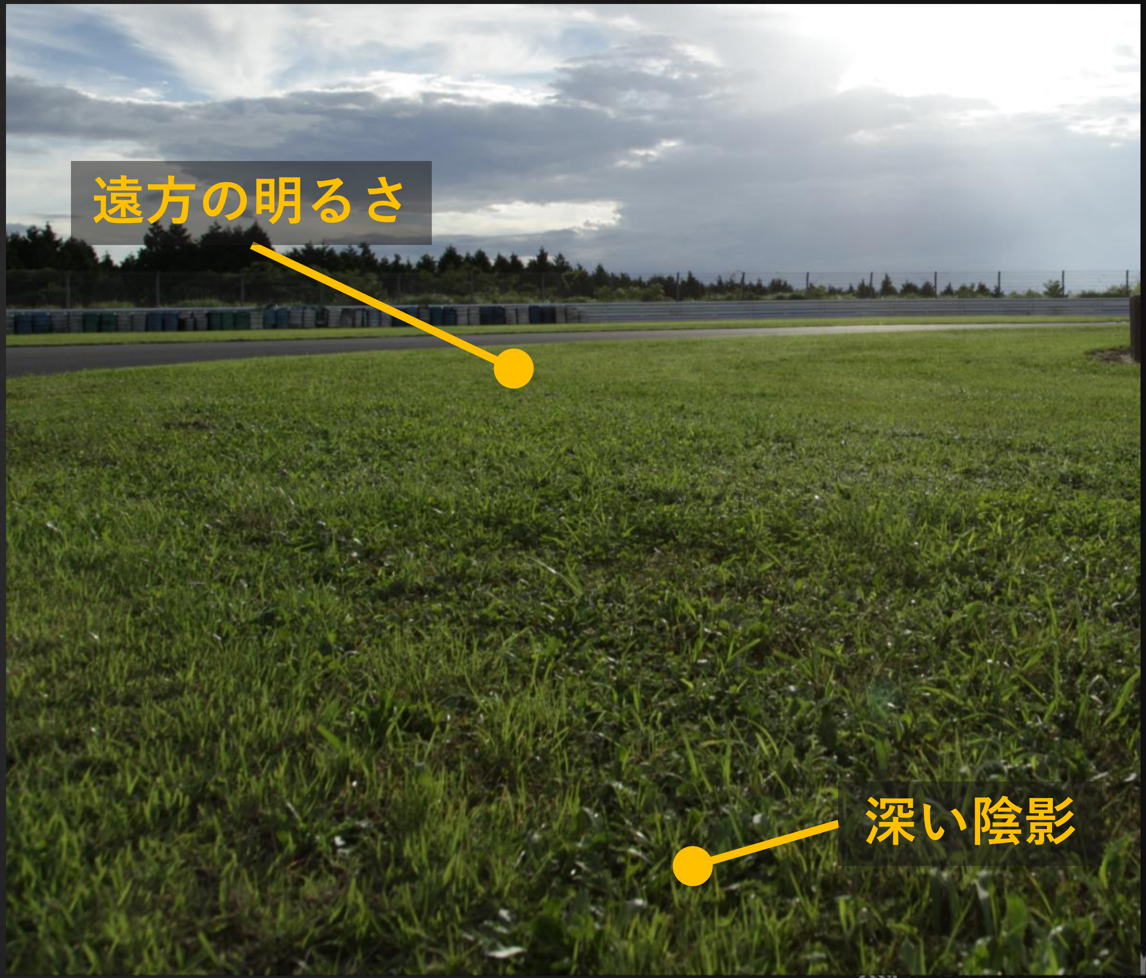
- ただし、**省いたものが場合によっては重要だったりも**する

→ 特殊な構造の物体には専用のシェーディングモデルを用いる
（場合がある）

POLYPHONY™ DIGITAL

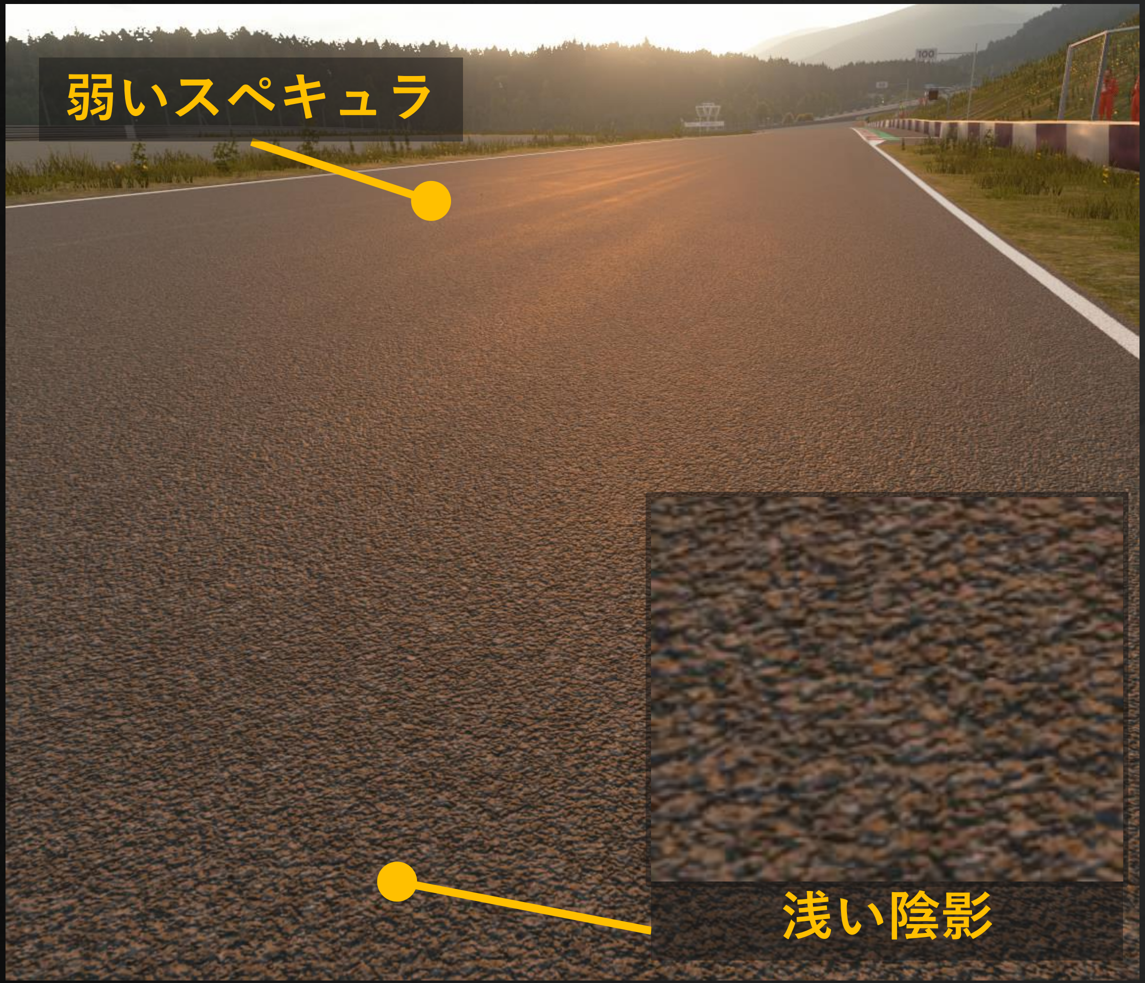
実写での見た目

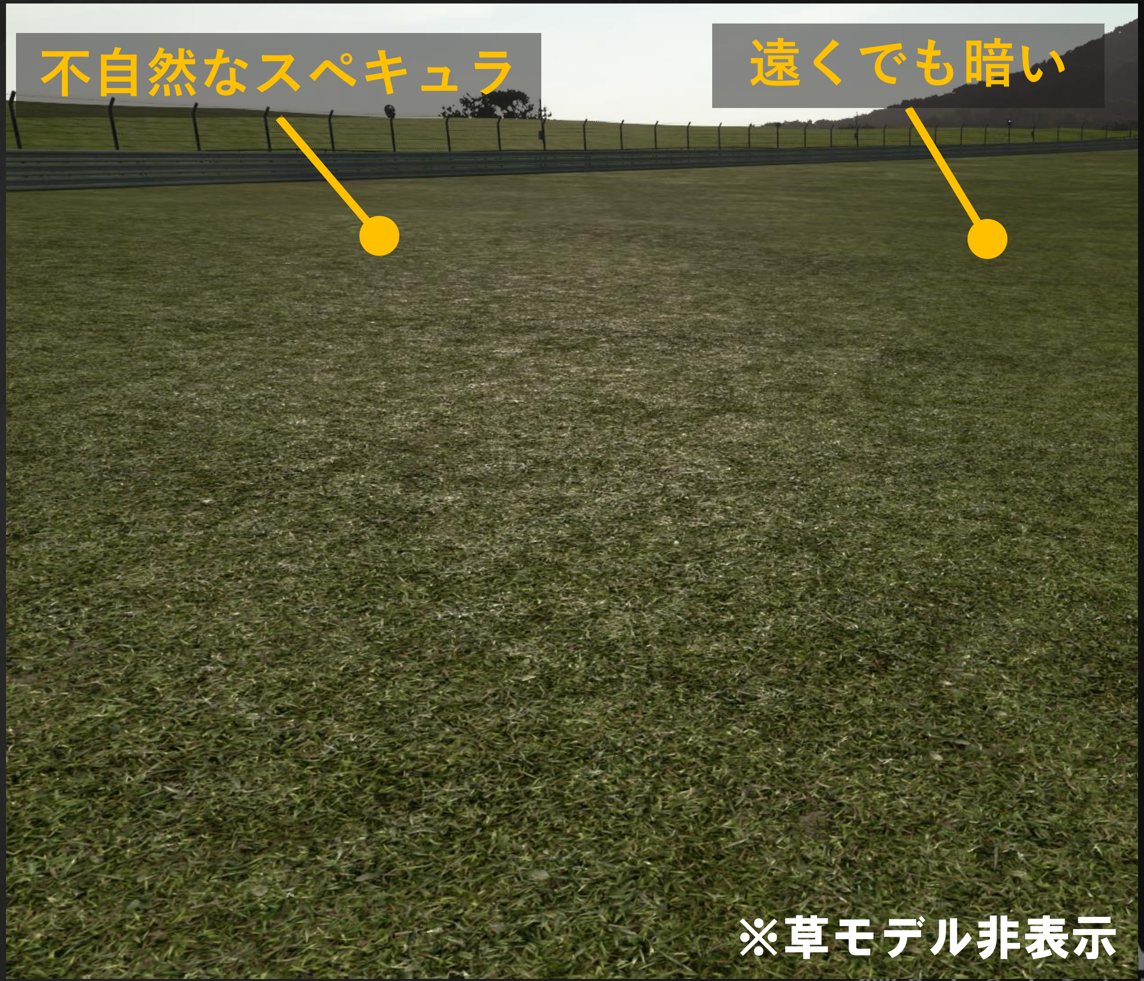
実写の路面・芝生には特徴的な見た目がある。



ゲーム内での見た目

ゲーム内の路面・芝生だとこうなりがち。（GT Sportでの見た目）



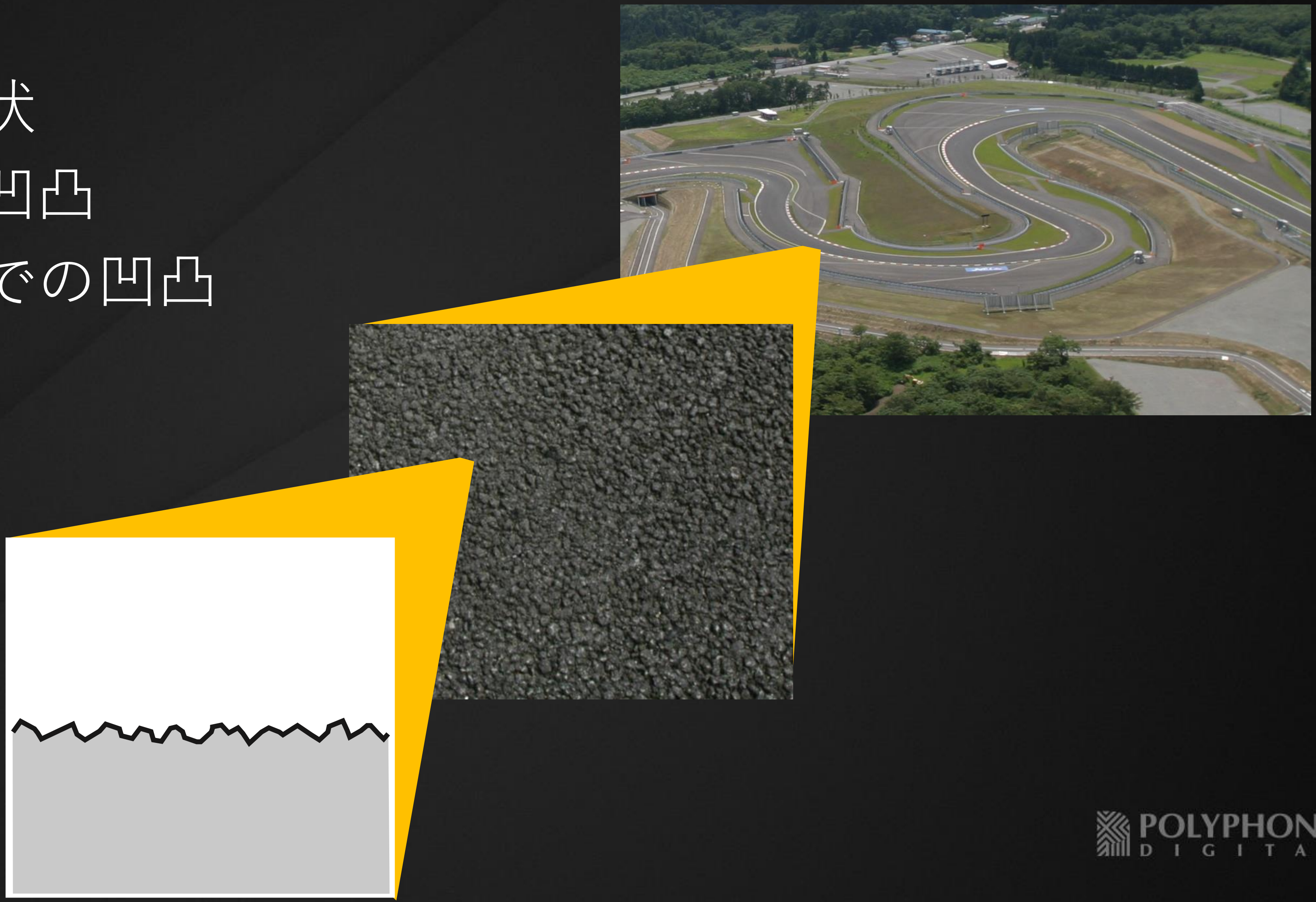


どうして再現が難しいのか

現実の物体形状は複雑

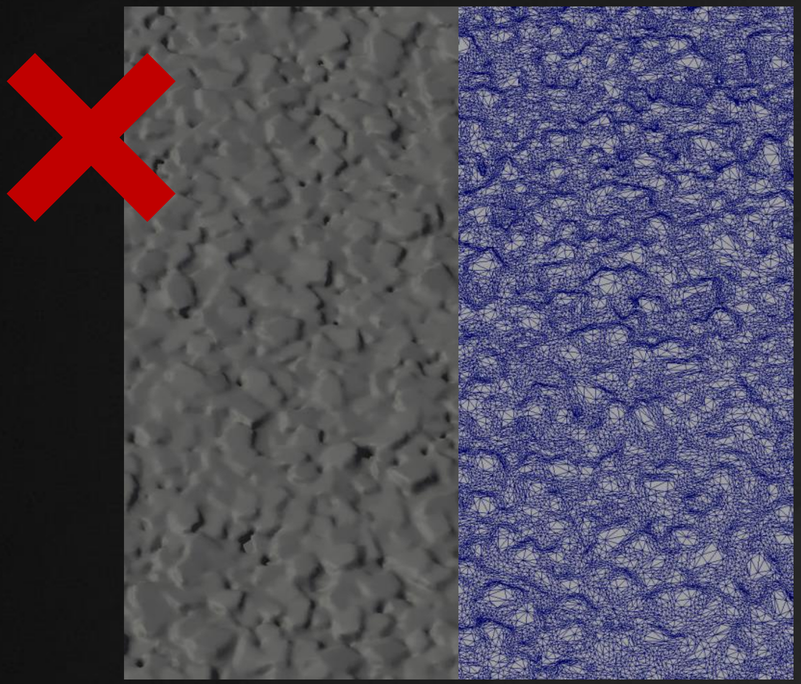
路面を例に

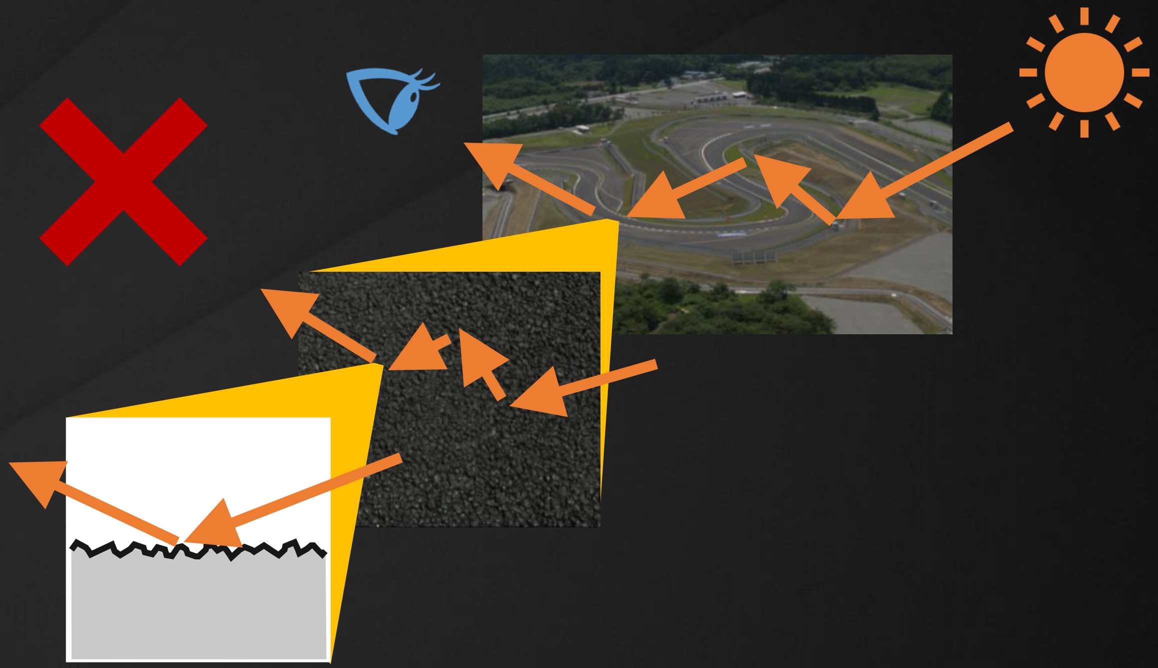
- サーキットの形状
- 砕石一個一個の凹凸
- 砕石一個の表面での凹凸



現実の物体形状は複雑

そのまま扱うことは出来ない。





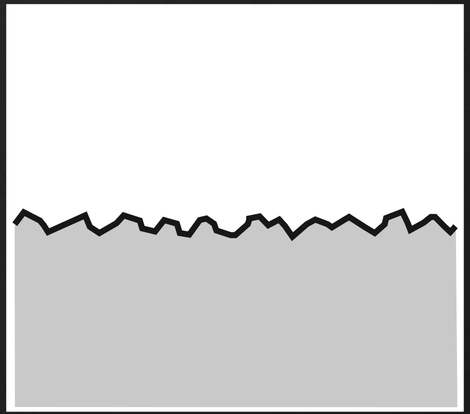
→一般的な手法ゲームではスケールごとに違う表現、計算

POLYPHONY DIGITAL

CGにおける形状表現

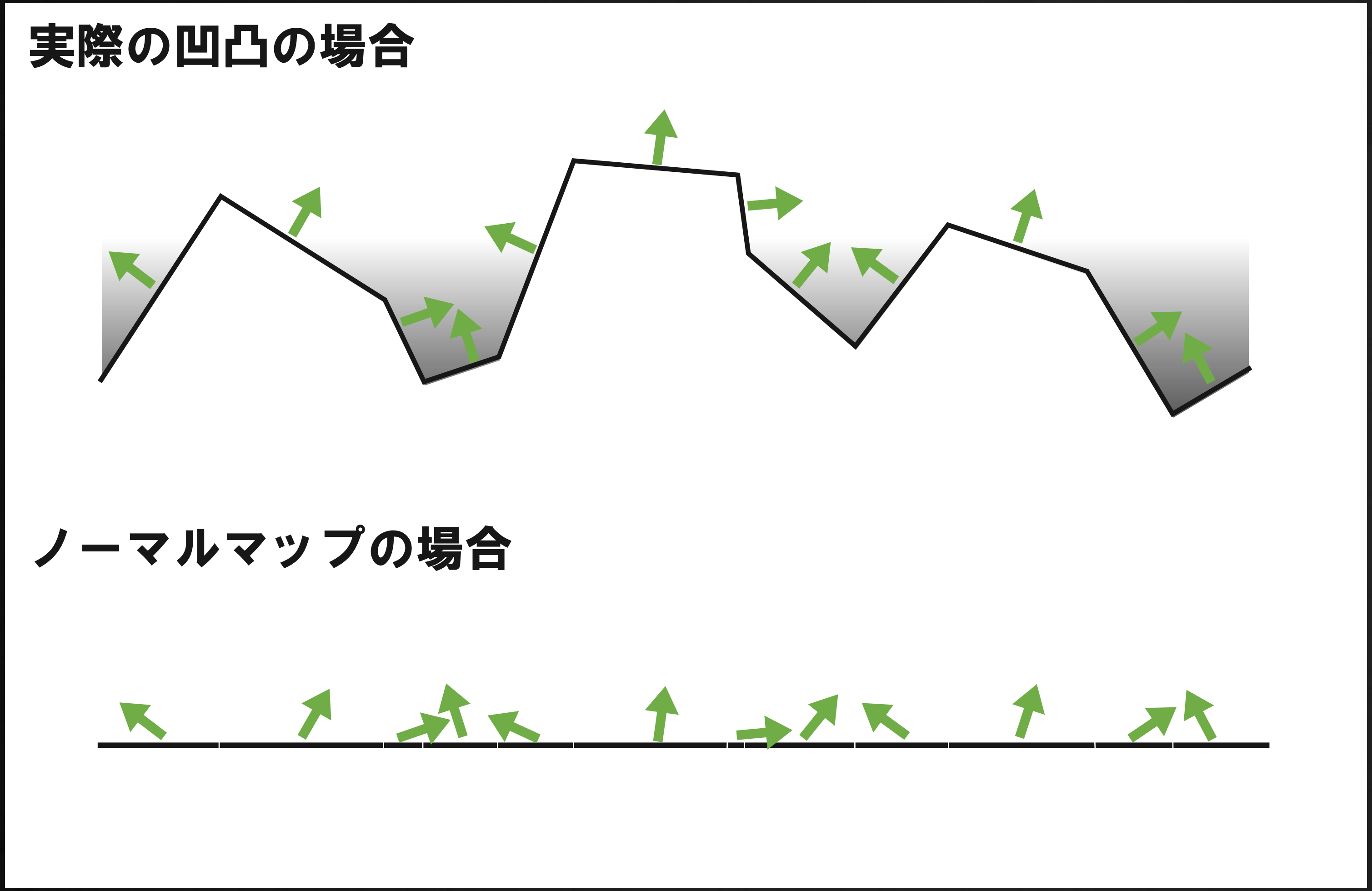
様々なスケールでの表現、計算

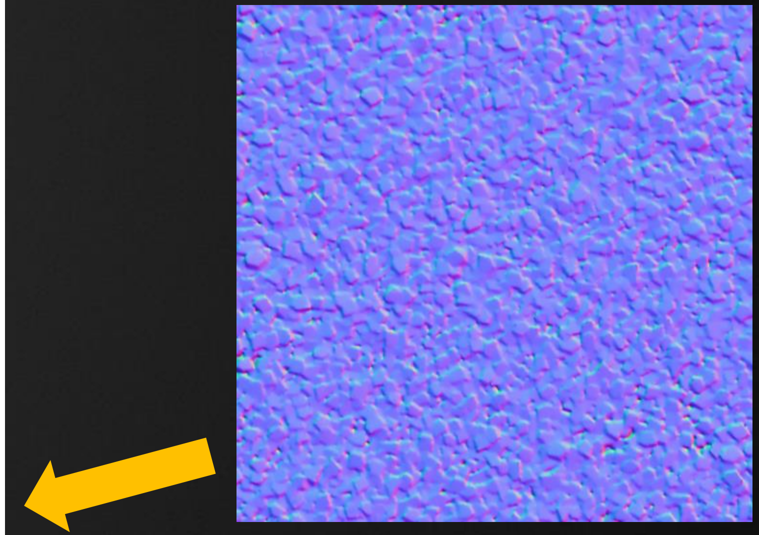
- **マクロスケール**
  - ポリゴンで表現される形状
- **メソスケール**
  - ノーマルマップなどの
    テクスチャで表現される形状
- **マイクロスケール**
  - ラフネスで表現される形状



ノーマルマップとは

ポリゴン上に貼ったテクスチャで法線だけ曲げる。



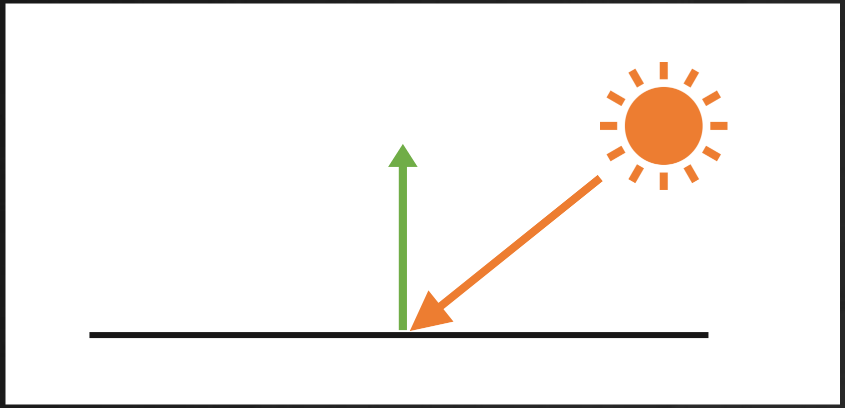


POLYPHONY DIGITAL

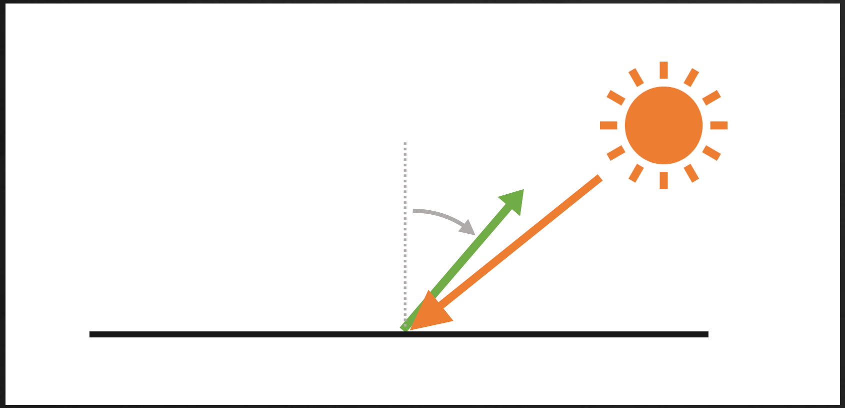
ノーマルマップとは

法線を曲げることで凹凸が存在するような見え方になる。

ノーマルマップなし



ノーマルマップあり

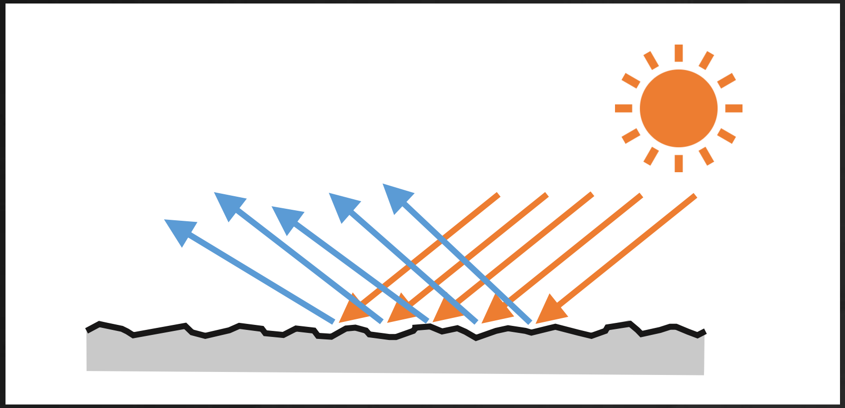


POLYPHONY™ DIGITAL

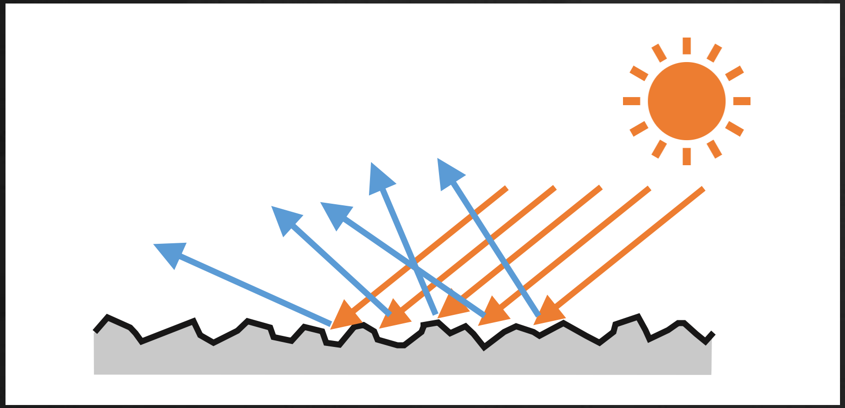
ラフネスとは

物体表面の微小平面の乱雑度が高いほど反射がボケる

低ラフネス



高ラフネス



POLYPHONY™ DIGITAL

ノーマルやラフネスによる表現は万能か？

- ノーマルやラフネスでの表現は大部分の場合では上手くいくが**失われる情報もある**

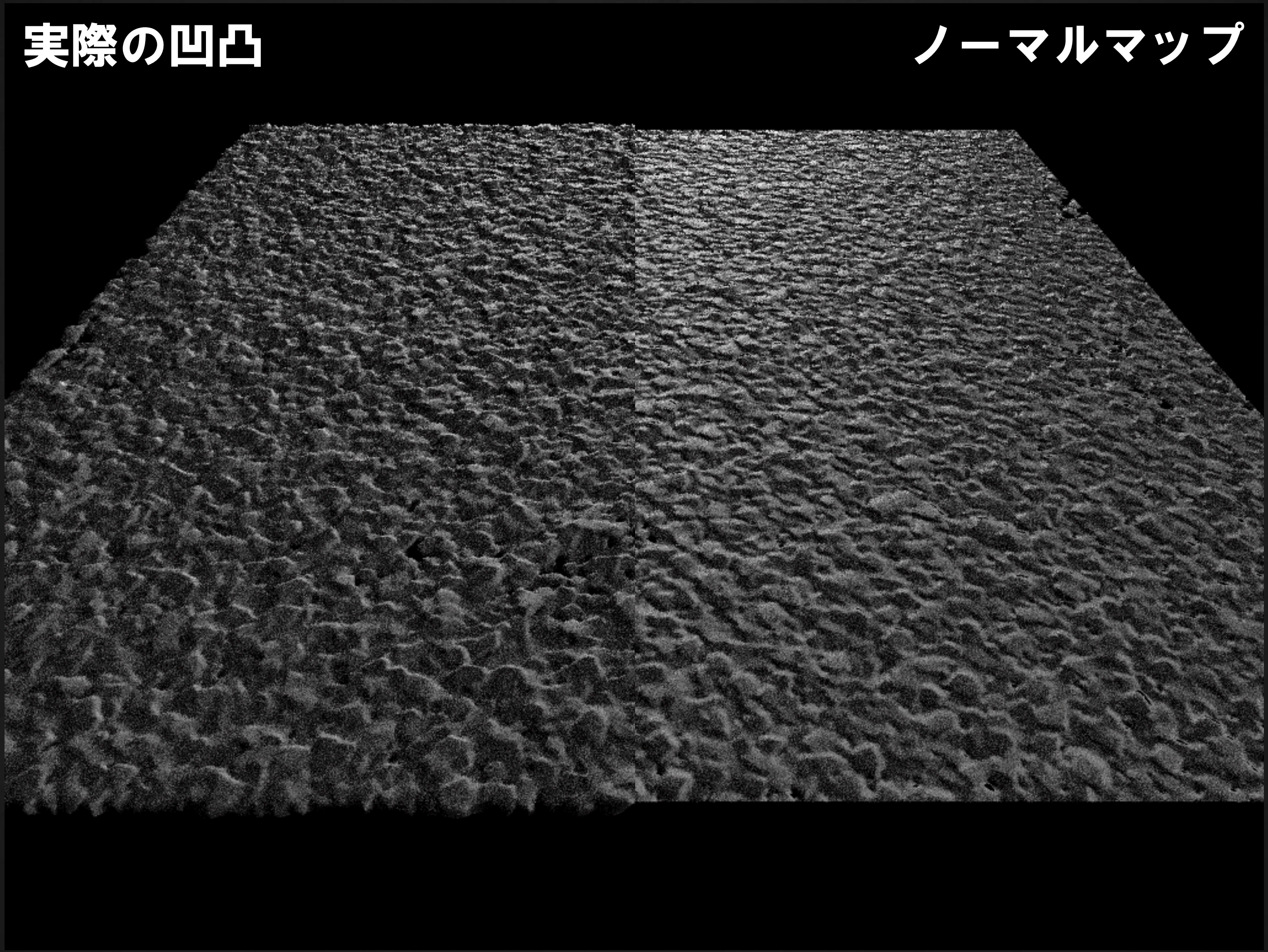
→ 今回は特にメソスケールをノーマルマップで表現することで失われる情報について着目

- 路面や芝生の表現に対して影響を与えている

POLYPHONY™ DIGITAL

実際の凹凸とノーマルマップの違い

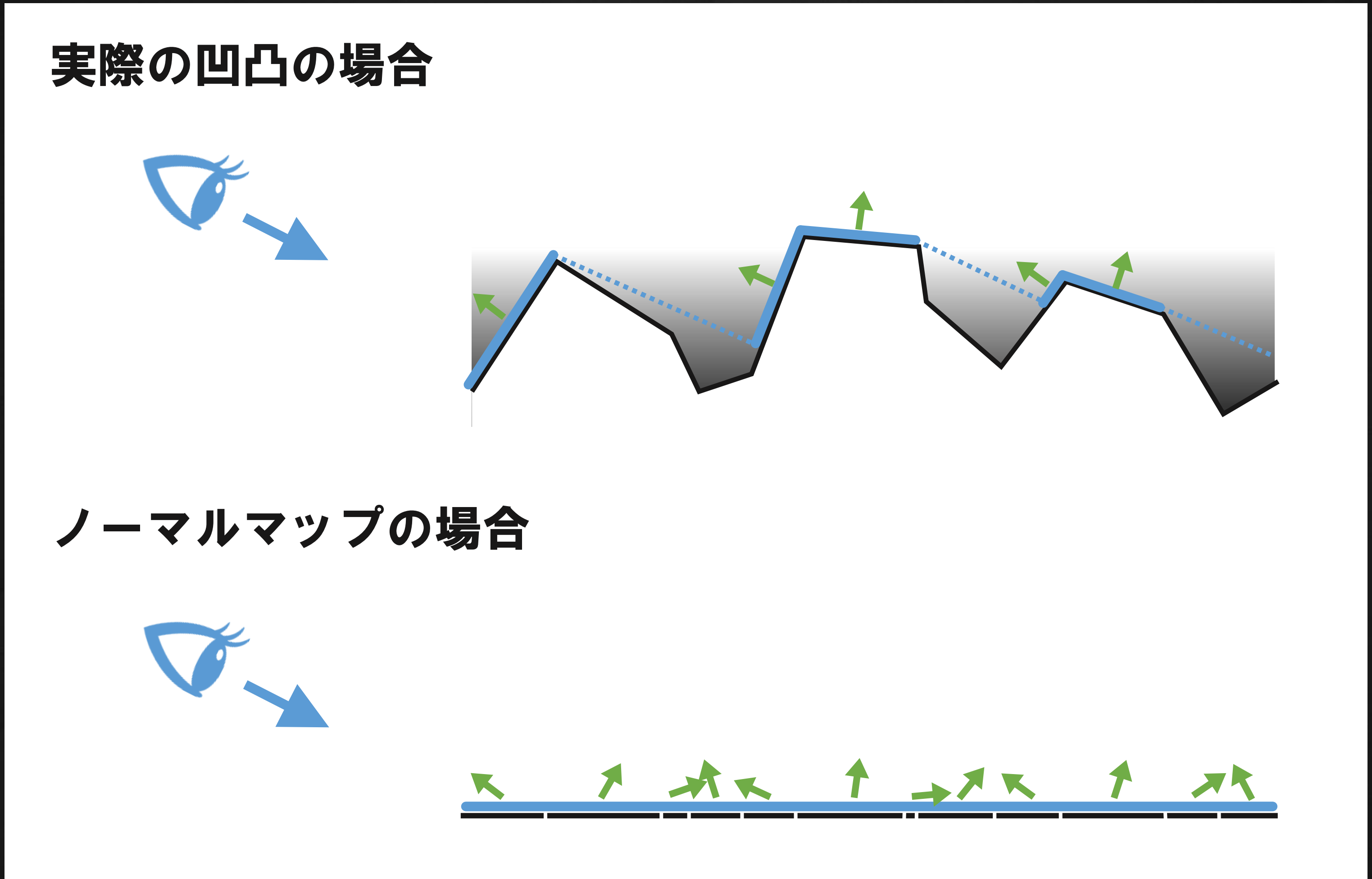
ノーマルマップは**実際の凹凸と見た目が変わる**場合がある。



POLYPHONY™ DIGITAL

視差

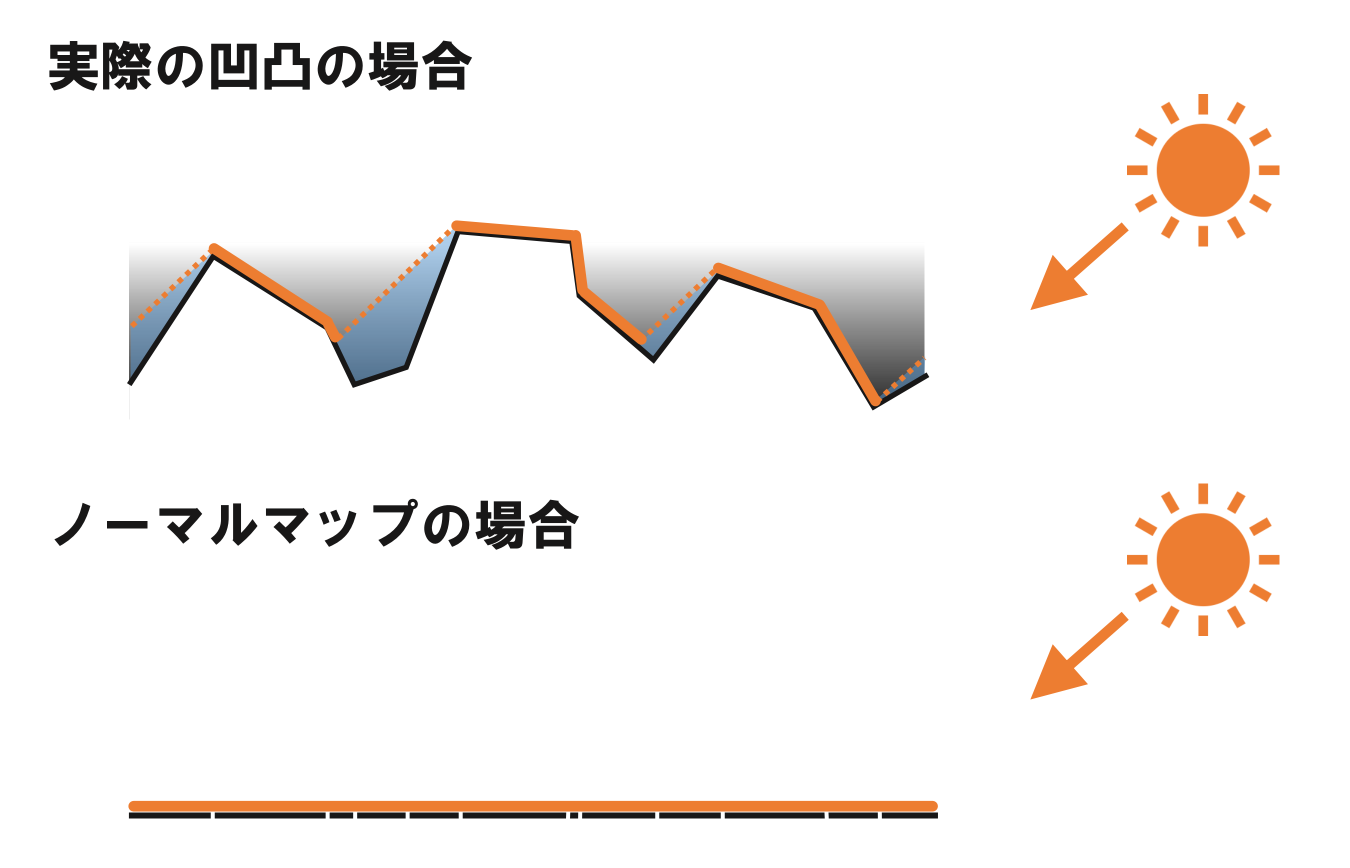
視線に対する遮蔽によって見える面・見えない面が出てくる現象



POLYPHONY™ DIGITAL

ローカルシャドウ

光に対する遮蔽によって**落ちる影**



※多重散乱については割愛

POLYPHONY™ DIGITAL

アドホックな対応

GT Sport 時点で取られていた対策

- 視差の再現
  - ラフネスを弱める
- ローカルシャドウの再現
  - ノーマルマップを強調する
  - ベースカラーのコントラストを強める

あちらを立てればこちらが立たずになりがち
→ 時間・天候変化のある GT7 では避けたい

Parallax Occlusion Mapping (POM) による表現

POMによる表現は必要な要件を満たさなかった。

- POMによる凹凸表現
  - 負荷の高い処理である。
  - 芝生のような高周波でオーバーハングの多い形状は難しい。
  - 遠方ではMipMapによって消えてしまう。
    - 広域的な見た目が表現出来ない。

→　PS5のみで一部の地面に使用するなどに留まる。

POLYPHONY™ DIGITAL

今回開発したシェーダー

路面・芝生シェーダーの開発

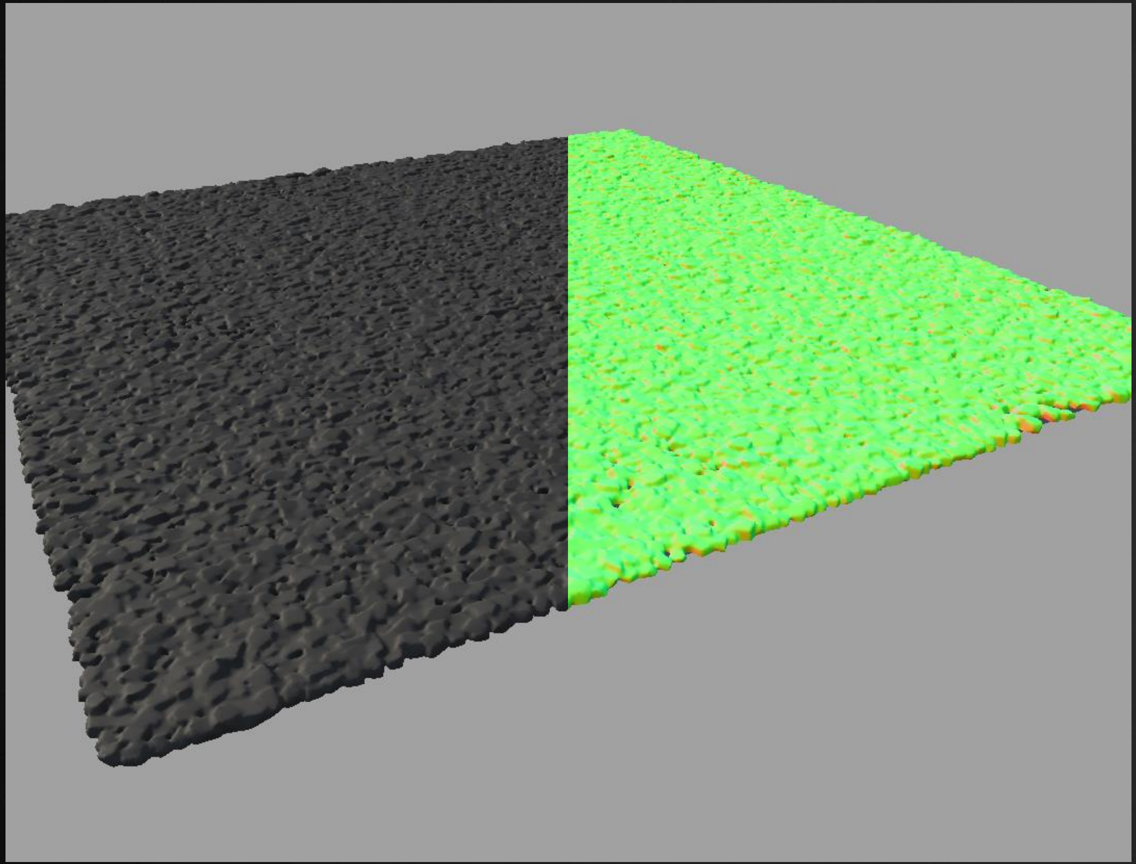
- **必要な要件と対する方針**
  - メソスケール構造による特徴的な見た目を再現できること。
    - **視差、ローカルシャドウ**の再現。
  - 近景と遠景両方で正しく表示できる。
    - MipMapにも対応する。
  - 時間変化・天候変化に対しロバストであること。
    - 物理ベースのアプローチを行う。
  - PS4上でも動作可能なほどランタイムの動作がシンプルであること。

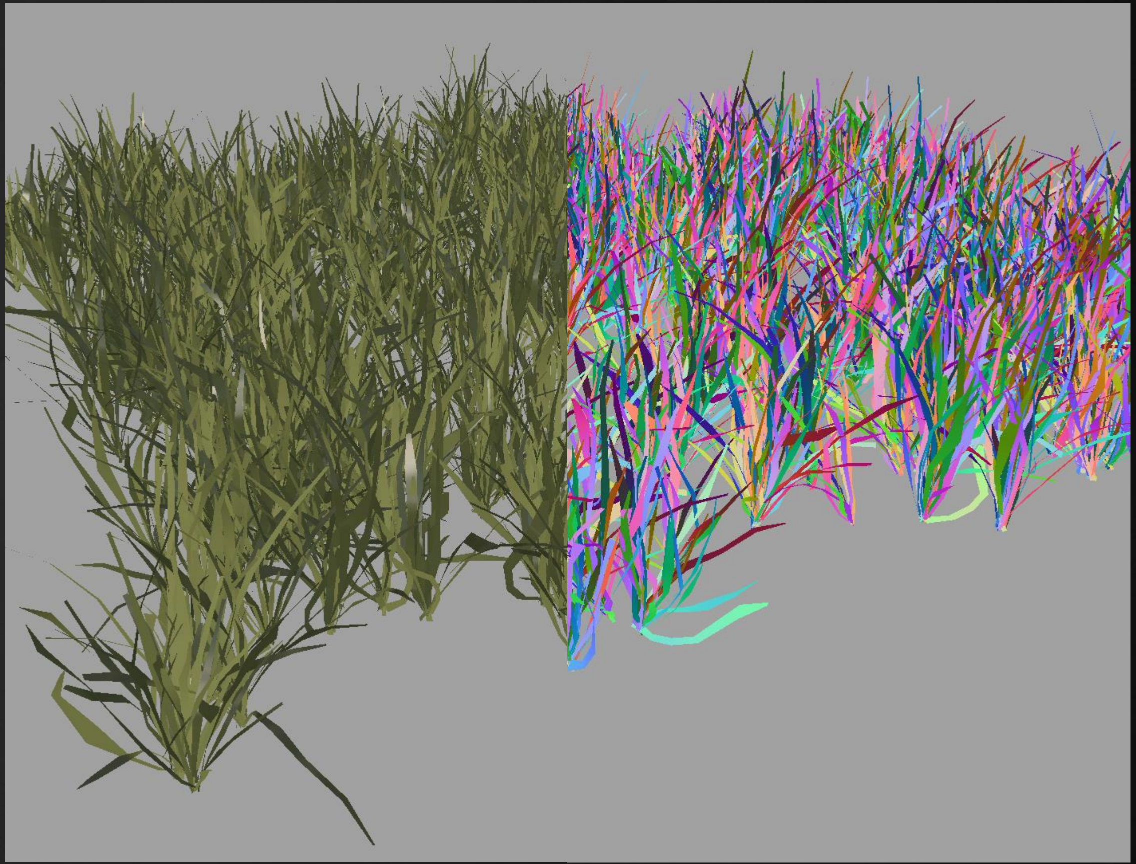
今回採用した手法

実際の凹凸から視差とローカルシャドウを測定して、
差分をノーマルマップに適用する
[Wu et al. 2019] から着想。

３Ｄモデルを用意

実際にメソスケール構造を持ったリファレンスモデルを作成。



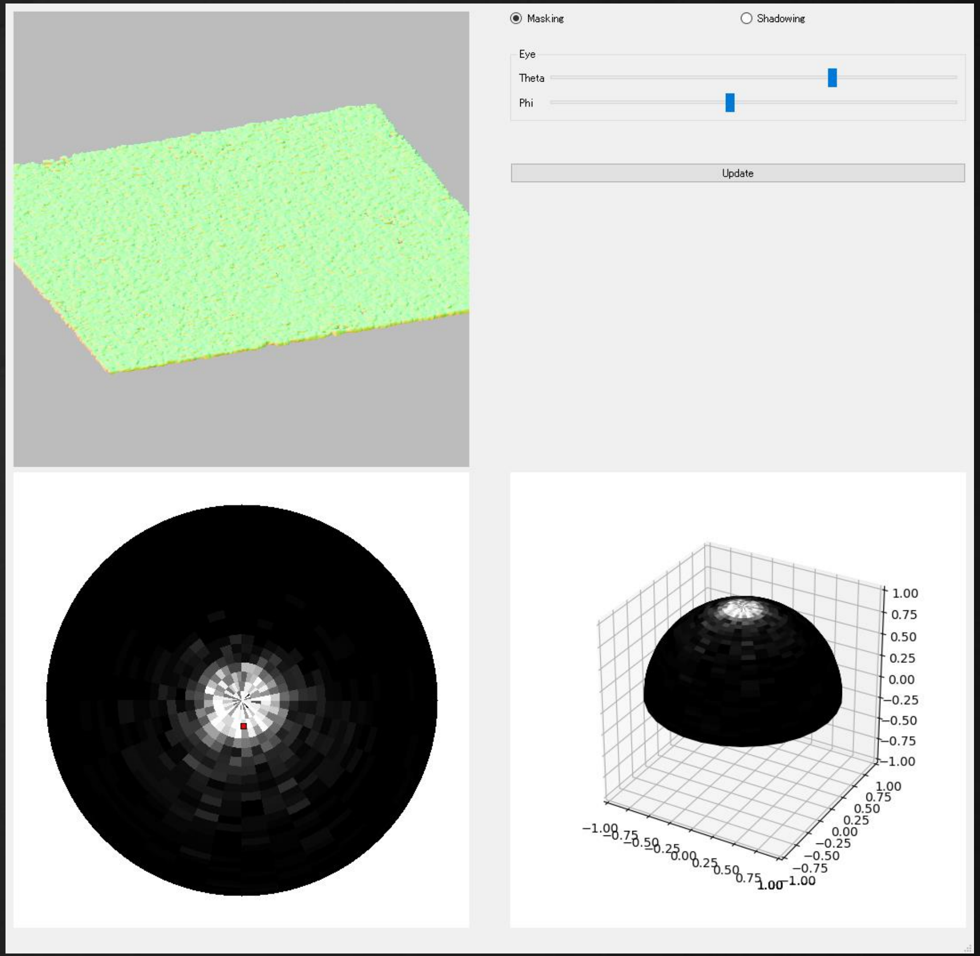


POLYPHONY DIGITAL

測定ツールを用意

視線方向や光源方向を変えつつ、
視差による変化やローカルシャドウ分布を計測するツールを作成した。

- 具体的に何を何のために測ってるかは後述
- 社内のレイトレ用フレームワークを活用 [Ozawa 2018]



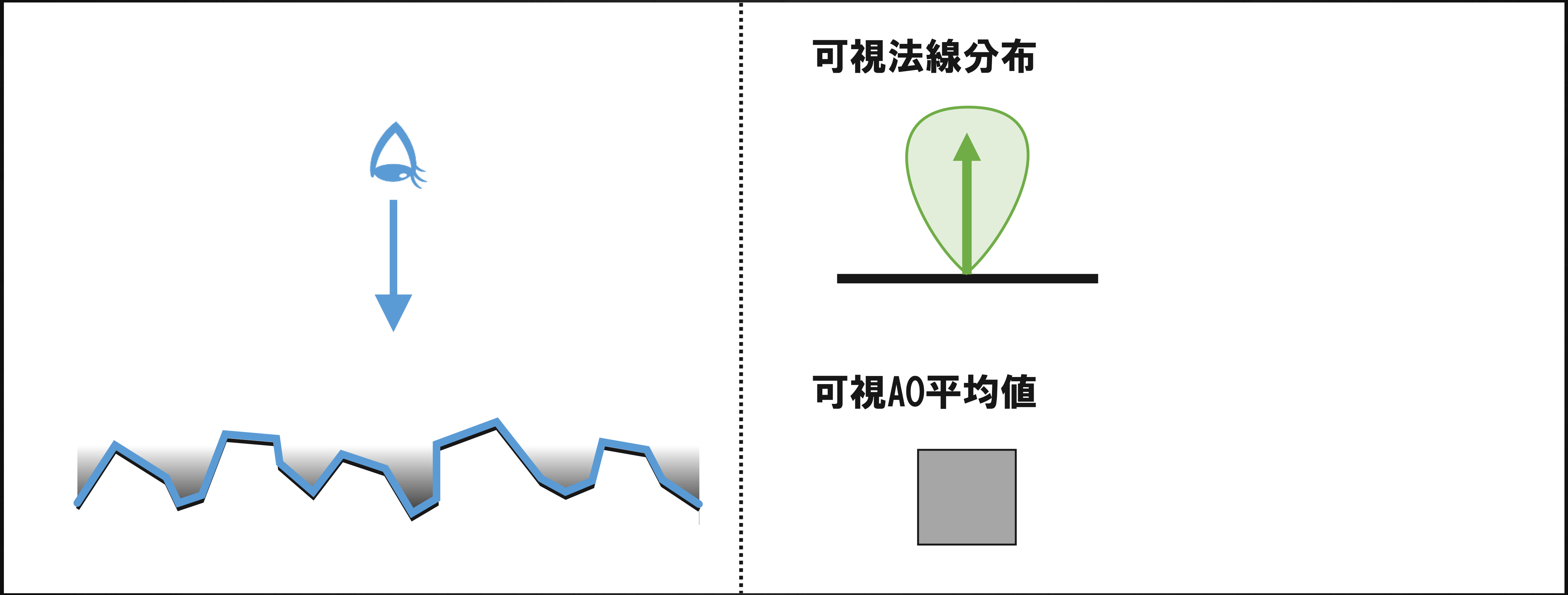
POLYPHONY™ DIGITAL

視差

POLYPHONY™ DIGITAL

視差：定性的な説明

視線方向が傾くと見かけ上のパラメーターが変わる。

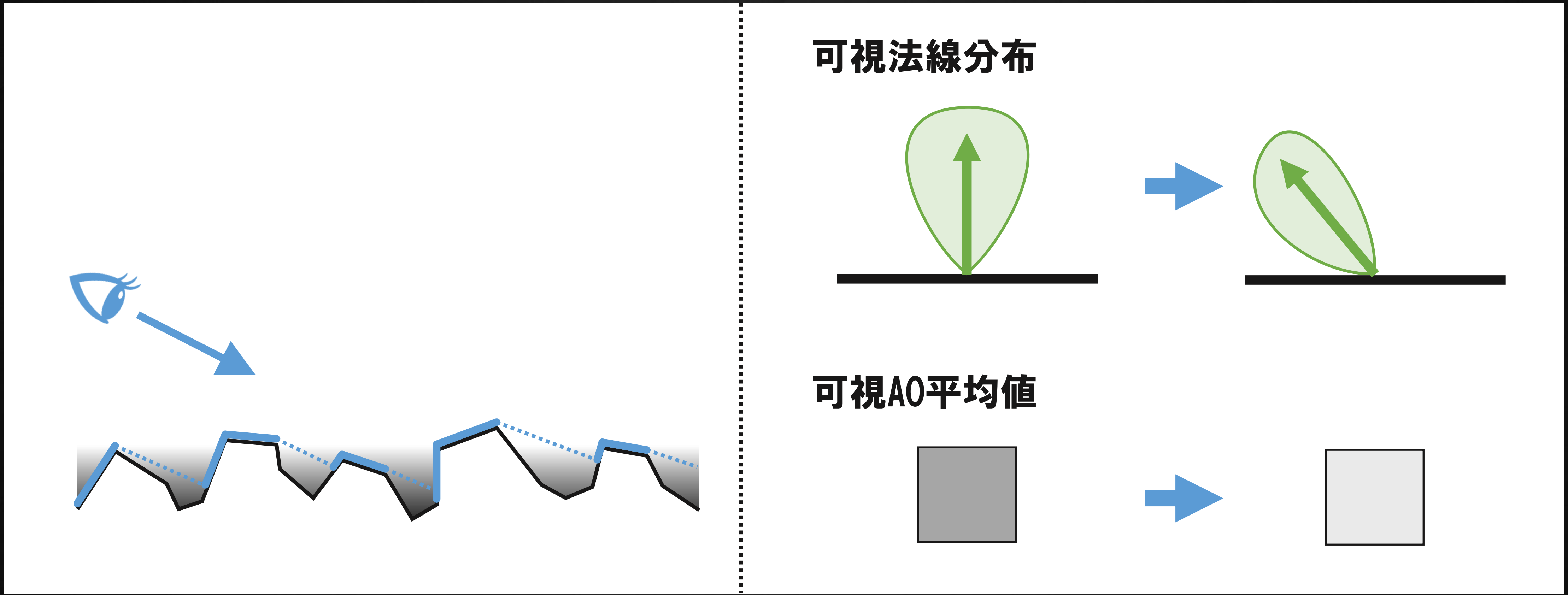


※AO（Ambient Occlusion）：ある点が周囲の形状から受ける遮蔽の程度

POLYPHONY™ DIGITAL

視差：定性的な説明

視線方向が傾くと見かけ上のパラメーターが変わる。



※AO（Ambient Occlusion）：ある点が周囲の形状から受ける遮蔽の程度

POLYPHONY™ DIGITAL

視差：モデル化

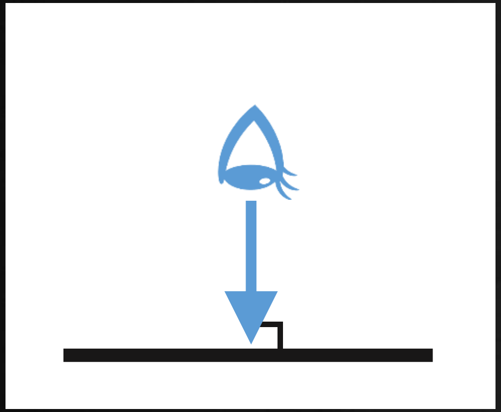
- 視線方向が傾いた際、下記のように変化させることにした。
  - ノーマルを視線方向に傾ける
  - ラフネスを低下させる
    - View-dependent roughness [Kojima et al. 2013] [Lagarde 2017]
  - AOを高くする

- レイトレでの**リファレンスと比較**し、
  どのパラメーターを変化させるべきか検証した。
  - ※場合によっては他のパラメーターを変化させる必要もある。
    あくまで上記は測定した形状についての結果。

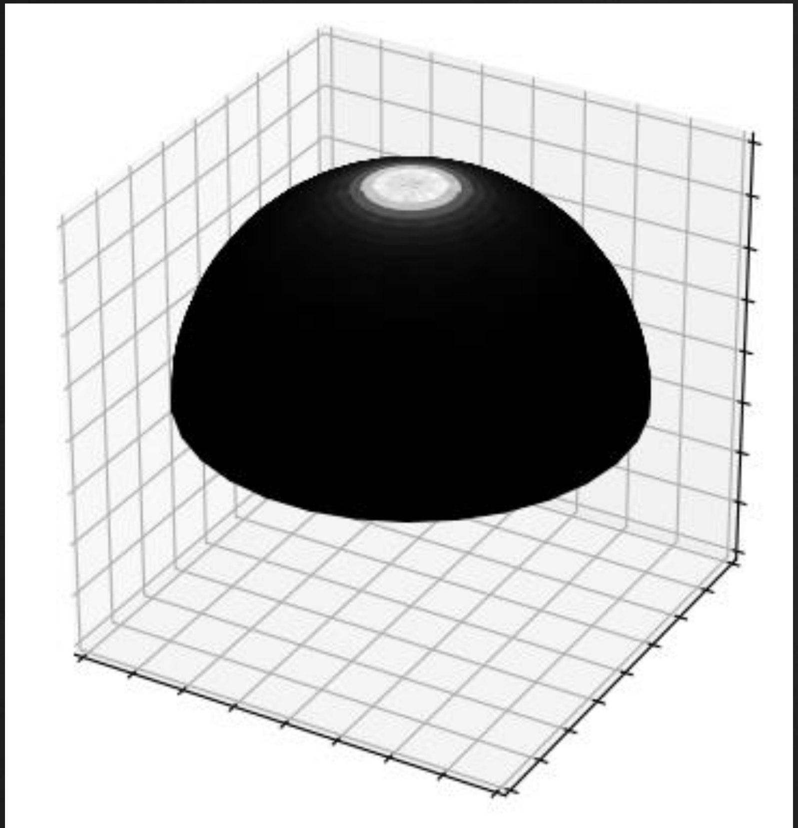
視差：変化の計測

例：可視法線分布の変化

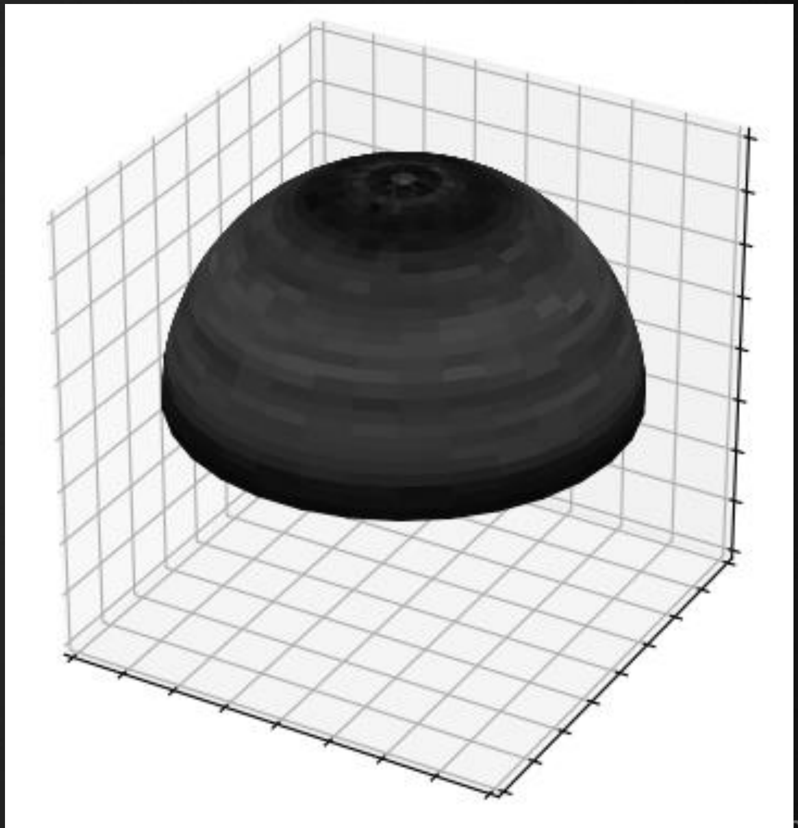
0°



路面



芝生

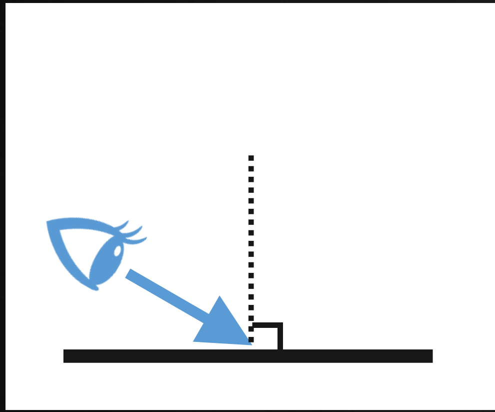


POLYPHONY™ DIGITAL

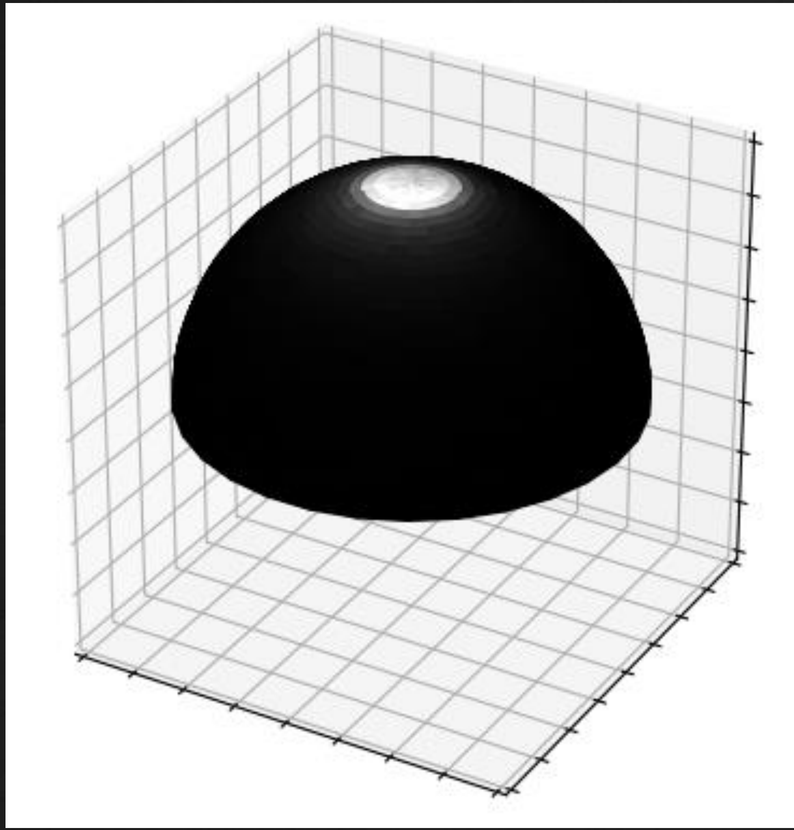
視差：変化の計測

例：可視法線分布の変化

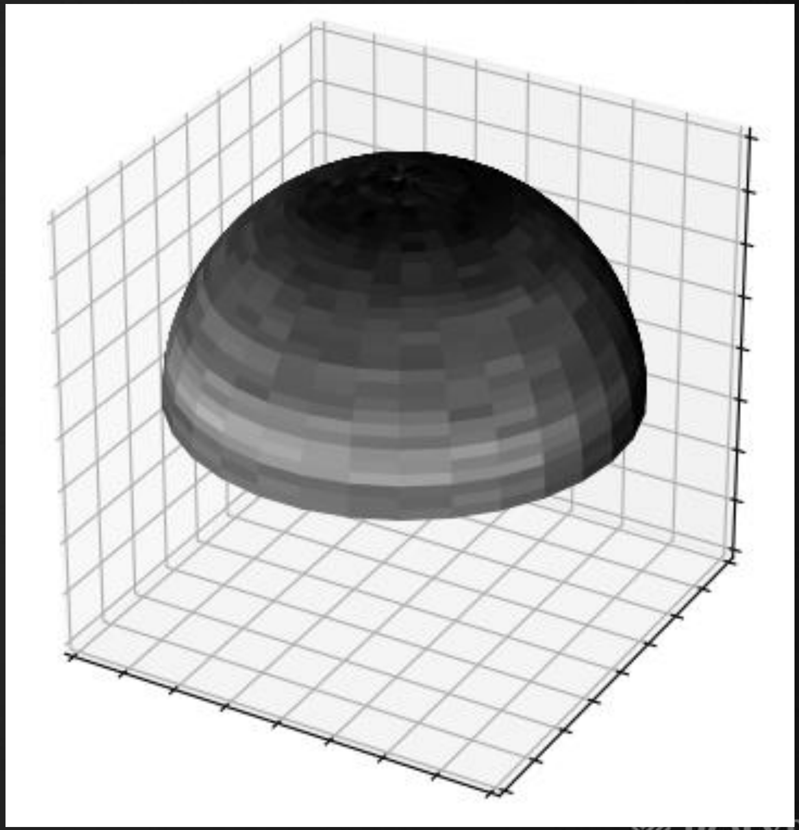
６０°



路面



芝生

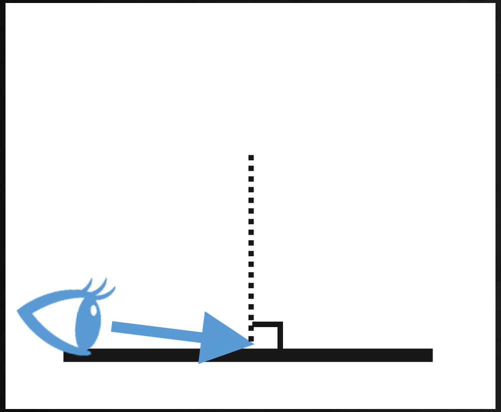


POLYPHONY™ DIGITAL

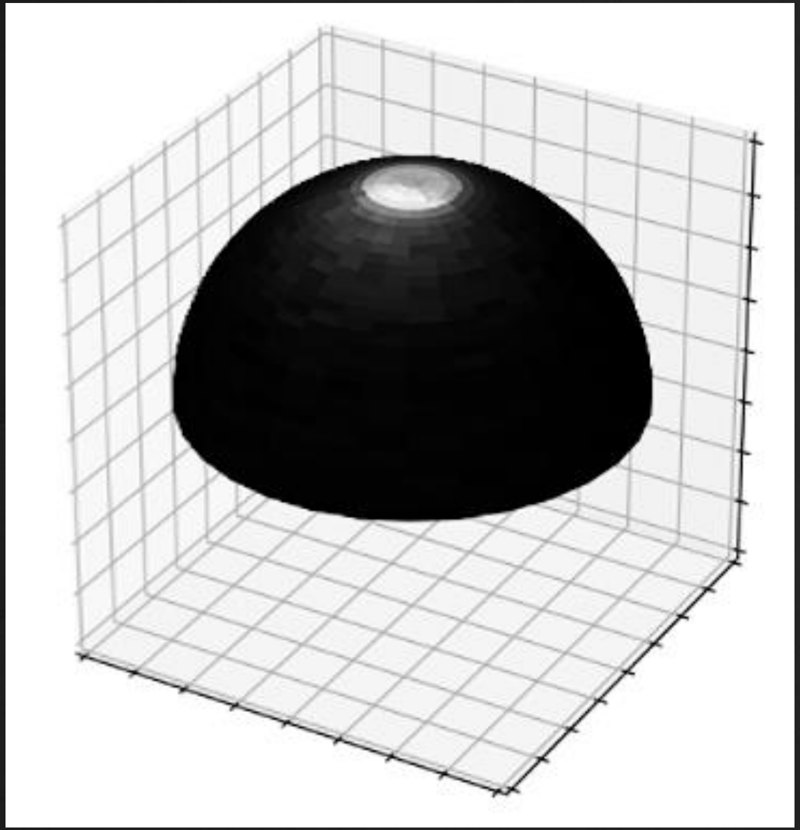
視差：変化の計測

例：可視法線分布の変化

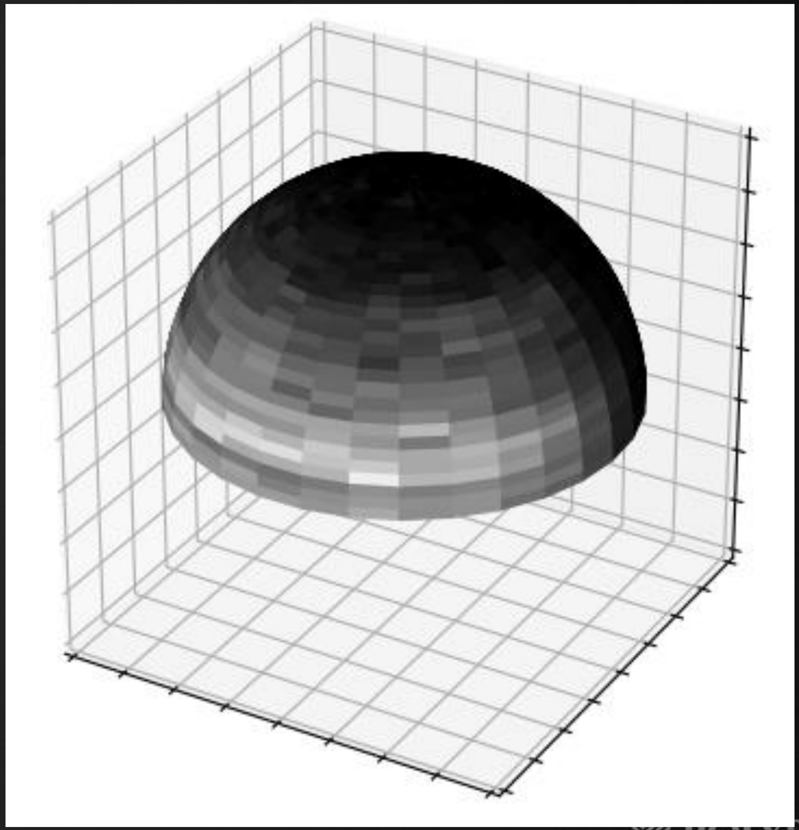
≒９０°



路面



芝生



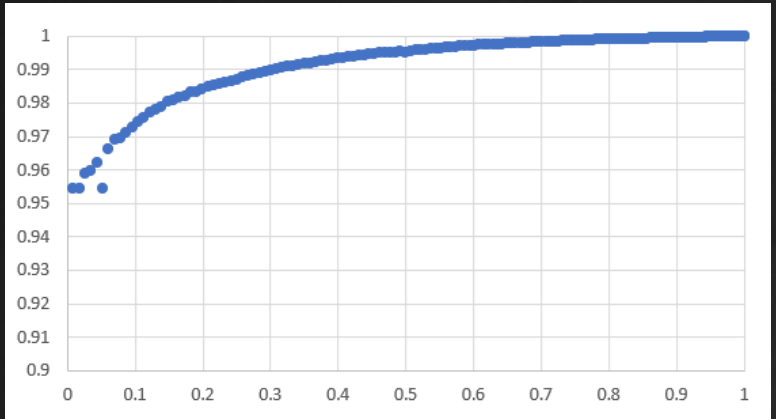
POLYPHONY™ DIGITAL

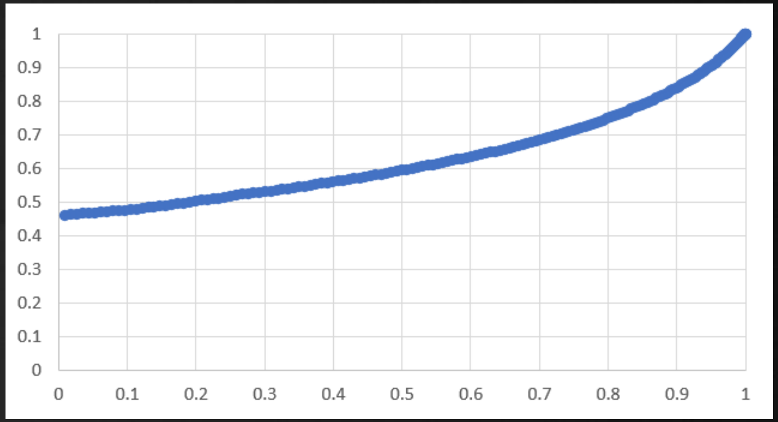
視差：計測結果

計測した変化を再現するような式をシェーダー上に実装した。

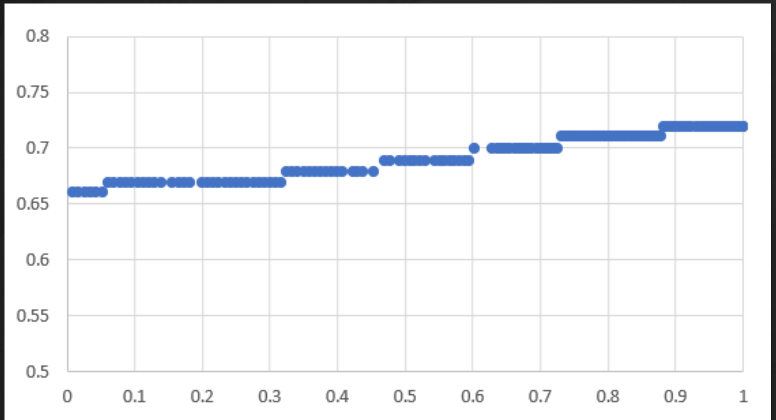
路面　　　　芝生

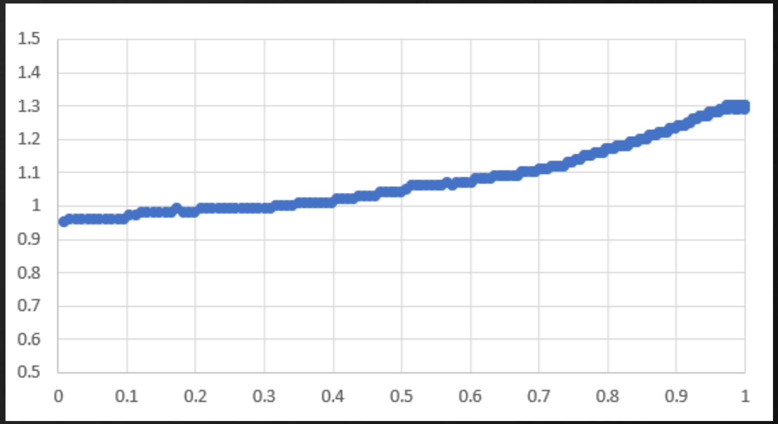
可視法線平均方向
$dot(N_{shading}, N_{geometry})$





ラフネス





視線方向
$dot(V, N_{geometry})$

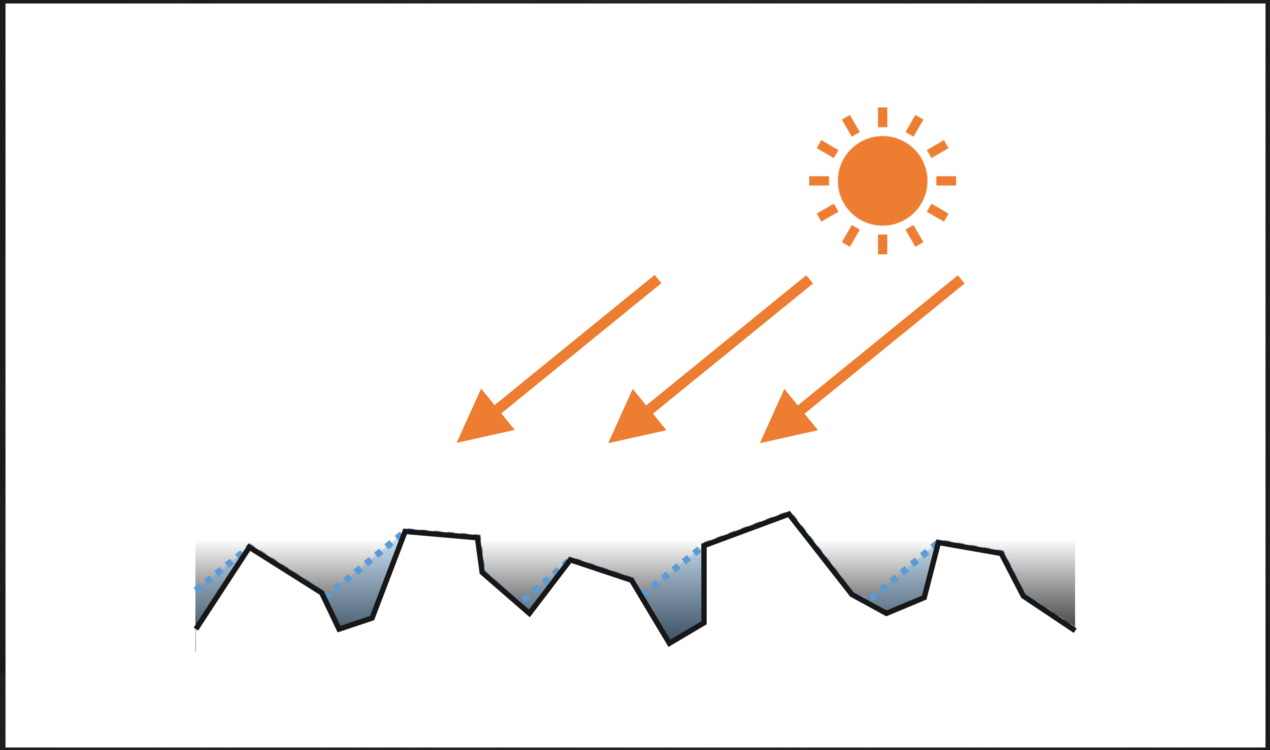
視線方向
$dot(V, N_{geometry})$

POLYPHONY DIGITAL™

ローカルシャドウ

ローカルシャドウ：定性的な説明

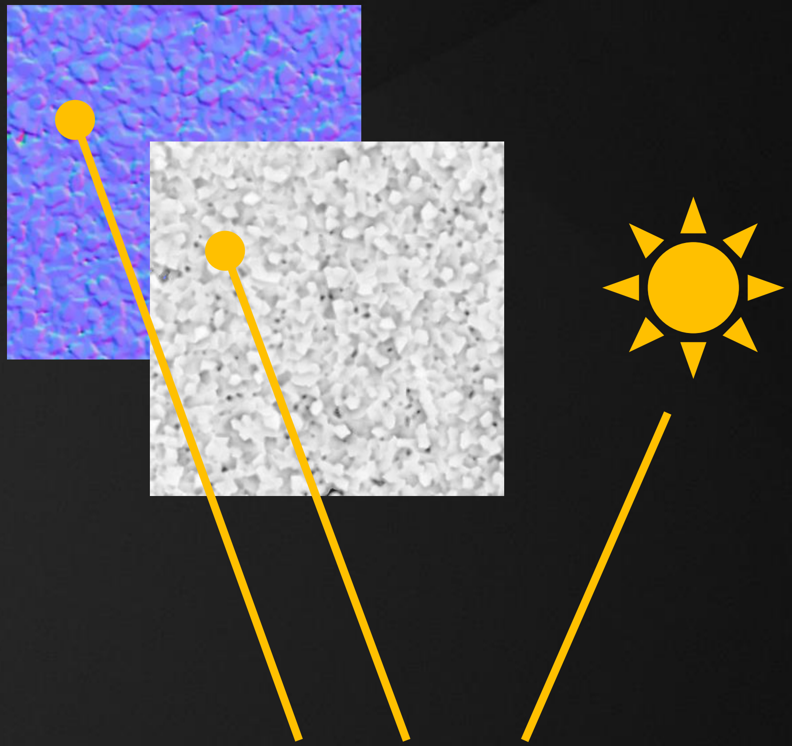
奥まっているところほど影になりやすい。



POLYPHONY™ DIGITAL

ローカルシャドウ：モデル化

- **メソスケール構造のAOを表すテクスチャ**を持たせる。
- 法線方向, AO, 光源方向 から**影の中かを近似的に判定する。**
  [Brinck et al. 2016]



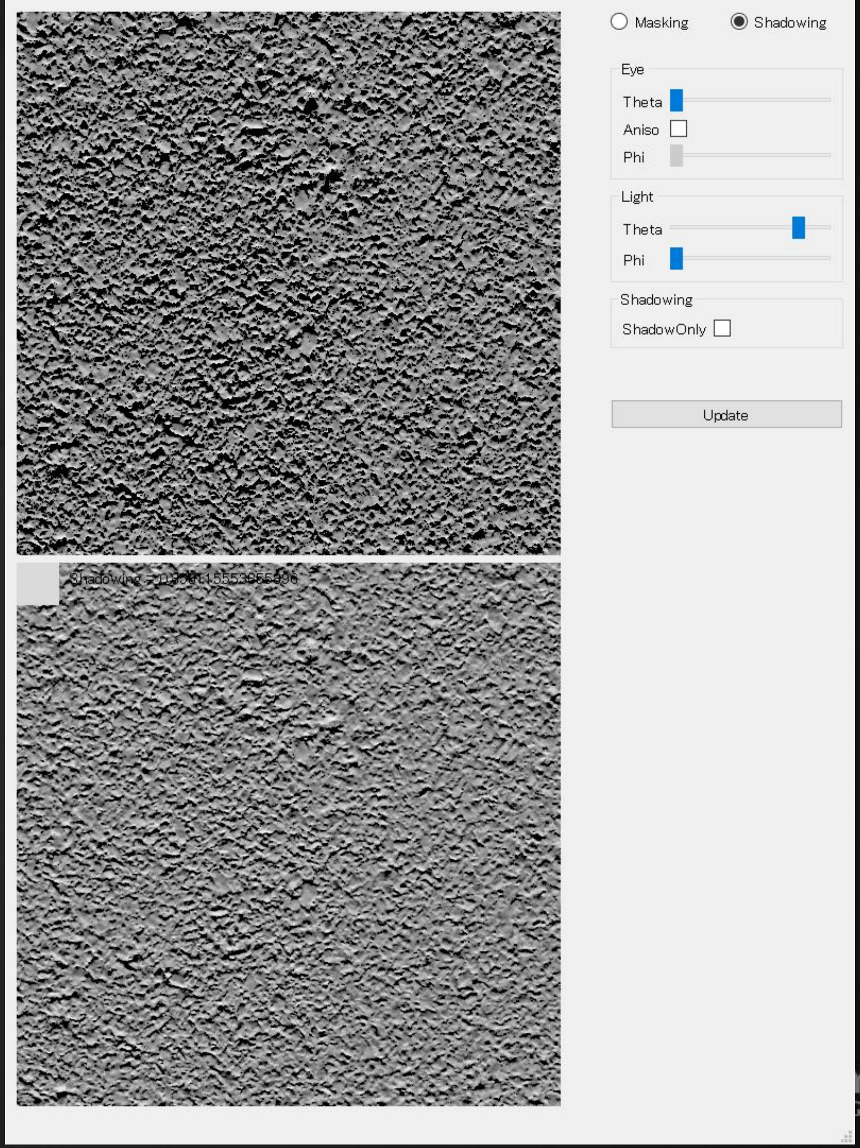
$$f_{shadow}(N, AO, L)$$

POLYPHONY™ DIGITAL

ローカルシャドウ：計測

例：路面のローカルシャドウ

- **レイトレーシングによる影と良く一致する影の式を探す。**
  - ツール上で、
レイトレーシングによる影と
近似式で出した影を比較。
  - 様々な式を試した。

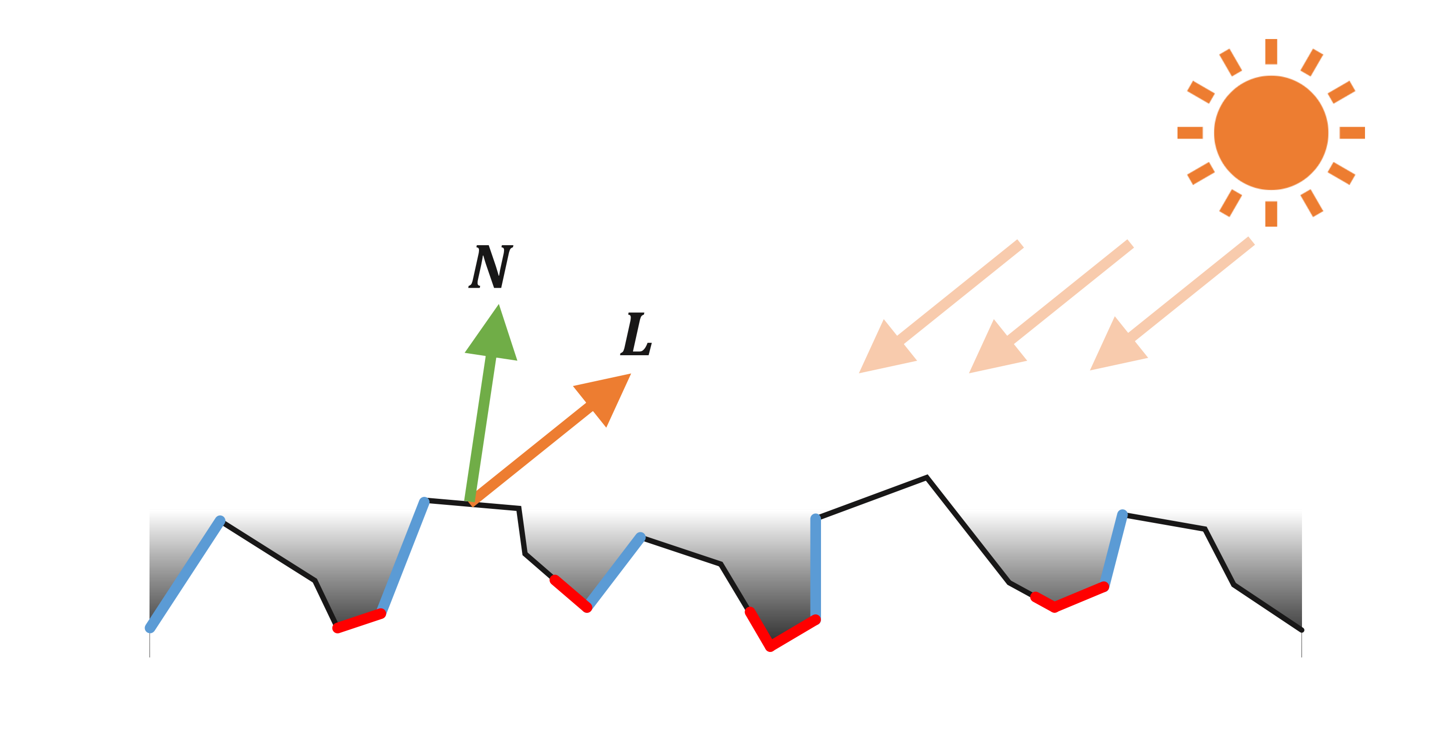


ローカルシャドウ：アイデア

例：路面のローカルシャドウ

- 光を受けない面を AO 値が低い所へ伸ばす。

$dot(N, L) < 0$ の面



POLYPHONY™ DIGITAL

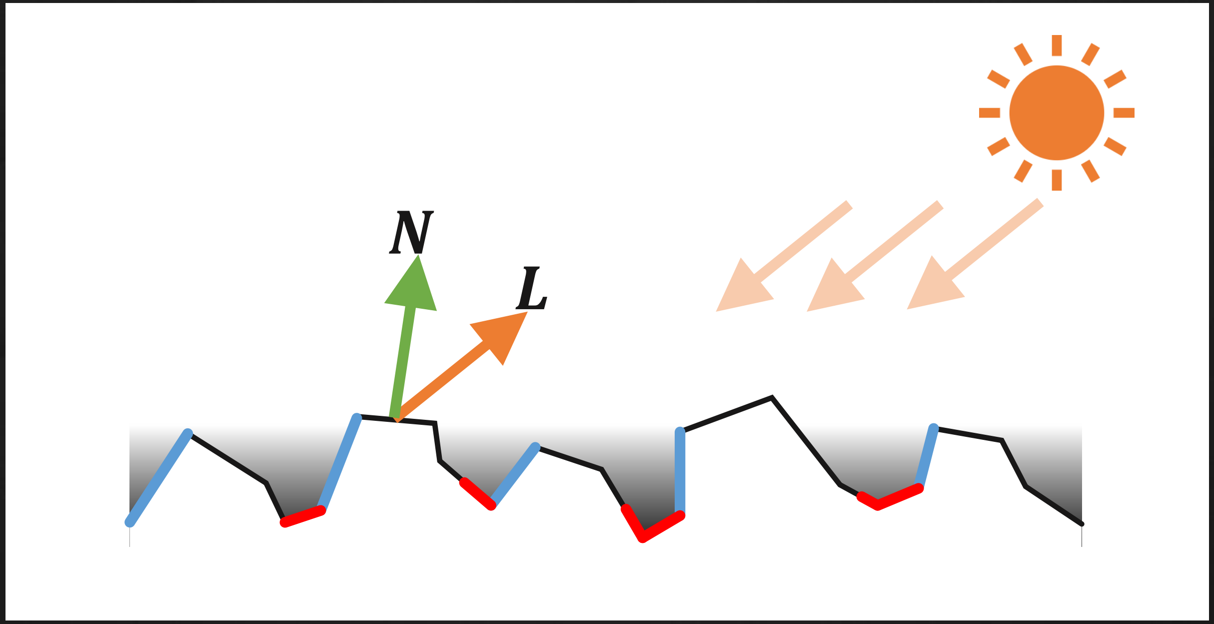
ローカルシャドウ：最終的な式

例：路面のローカルシャドウ

$$f_{shadow}(N, AO, L) = clamp01(dot(N, L) - \alpha(1 - AO) + 0.5)$$

α：定数（影の強さ）

MipMapのために出来るだけ線形になるように考慮。



POLYPHONY™ DIGITAL

結果

適用前

適用後

適用前

POLYPHONY™ DIGITAL

適用後

POLYPHONY DIGITAL

適用前：動画

適用後：動画

適用前
unicorn media
Transurban
MILES
FLY AMERICA
BRIGHTON ANTIQUES
BLU STRADA
ADORF
※草モデル非表示

適用後

unicorn media
Transurban
MILES
LEMPEL
FLY AMERICA
BRIGHTON ANTIQUES
BLU STRADA
ADORF

※草モデル非表示

適用前：動画

※草モデル非表示

適用後：動画

※草モデル非表示

見た目の一貫性

見た目の一貫性

見た目の一貫性

見た目の一貫性

今後

- 他の構造のマテリアルへの一般化
  - 土、木、布、…
- マクロスケールジオメトリとの理論的な統合
  - 見た目を変えずにLoDを変化させる
- 多重散乱のモデル化を行う

POLYPHONY™ DIGITAL

まとめ

- CGにおける材質や形状表現についての説明
- メソスケール構造の見た目への影響を考慮した路面・芝生のシェーダーを実装した。
  - **視差**
    - 可視法線分布や可視AOの変化を計測し、シェーダー上で再現した。
  - **ローカルシャドウ**
    - リファレンスとの比較を通して軽量な近似式を探し、実装した。

POLYPHONY™ DIGITAL

参考文献

- [Wu et al. 2019] Lifan Wu, Shuang Zhao, Ling-Qi Yan, and Ravi Ramamoorthi. 2019. Accurate appearance preserving prefiltering for rendering displacement-mapped surfaces. ACM Trans. Graph. 38, 4 (August 2019). DOI: https://doi.org/10.1145/3306346.3322936
- [Ozawa 2018] Yuki Ozawa. 2018. ゲーム開発のためのレイトレーシング. CEDEC 2018. https://cedil.cesa.or.jp/cedil_sessions/view/1894
- [Kojima et al. 2013] Hideo Kojima, Hideki Sasaki, Junji Tago, Masayuki Suzuki. 2013. MGS5 & Fox Engine. GDC 2013 Panel.
- [Lagarde 2017] Sébastien Lagarde. 2017. Physically-Based Materials: Where Are We? In ACM SIGGRAPH 2017 Courses (SIGGRAPH '17), Association for Computing Machinery, Los Angeles, California.
- [Brinck et al. 2016] Waylon Brinck, Andrew Maximov, Yibing Jiang. 2016. The technical art of uncharted 4. In ACM SIGGRAPH 2016 Courses (SIGGRAPH '16), Association for Computing Machinery, Anaheim, California. DOI: https://doi.org/10.1145/2897839.2936731

POLYPHONY™ DIGITAL

おわり

グランツーリスモ7のグラフィックス技術を紹介

- リアルタイムレイトレーシング
  - 物理ベースレンダリングについての簡単な解説
  - レイトレーシングを使ってスペキュラーGIを計算
  - 複数のフィルタによるデノイズでサンプルを削減
- 路面・芝生のメソスケール材質表現
  - CGにおける材質や形状表現についての説明
  - メソスケール構造の見た目への影響を考慮した路面・芝生のシェーダーを実装
    - 視差
    - ローカルシャドウ

POLYPHONY™ DIGITAL